This thesis is presented for the degree of Doctor of Philosophy of Murdoch University

Deep Learning Techniques for Image Captioning

Md. Zakir Hossain
Doctor of Philosophy
Information Technology, Murdoch University, Australia.

September 2020

Declaration

I declare that this thesis is my own account of my research and contains as its main content work which has not previously been submitted for a degree at any tertiary education institution.

.....
Md. Zakir Hossain

Acknowledgments

Firstly, I would like to express my sincere gratitude to my principal supervisor Associate Professor Ferdous Sohel for the continuous support of my PhD study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better supervisor and mentor for my PhD study.

Besides my supervisor, I would like to thank my co-supervisors Dr. Mohd Fairuz Shiratuddin, and Associate Professor Hamid Laga for their insightful comments, encouragements, and perspectives.

I would like to thank Winthrop Professor Mohammed Bennamoun from the University of Western Australia for his support as a mentor.

I would also like to thank my family: my parents and to my brothers and sister for supporting me. A special thanks to my wife Tanny, my two sons Sami and Fahim for their constant encouragement, support, and positive outlook.

This research is supported by Murdoch International Postgraduate Scholarship (MIPS). I am thankful to Murdoch University, Australia.

Abstract

Generating a description of an image is called image captioning. Image captioning is a challenging task because it involves the understanding of the main objects, their attributes, and their relationships in an image. It also involves the generation of syntactically and semantically meaningful descriptions of the images in natural language. A typical image captioning pipeline comprises an image encoder and a language decoder. Convolutional Neural Networks (CNNs) are widely used as the encoder while Long short-term memory (LSTM) networks are used as the decoder. A variety of LSTMs and CNNs including attention mechanisms are used to generate meaningful and accurate captions. Traditional image captioning techniques have limitations in generating semantically meaningful and superior captions. In this research, we focus on advanced image captioning techniques, which are able to generate semantically more meaningful and superior captions. As such we have made four contributions in this thesis.

First, we investigate an attention based LSTM on image features extracted by DenseNet, which is a newer type of CNN. We integrate DenseNet features with attention mechanism and we show that this combination can generate more relevant image captions than other CNNs.

Second, we use bi-directional self-attention as a language decoder. Bi-directional decoder can capture the context in both forward and backward directions, i.e., past context as well as any future context, in caption generation. Consequently, the generated captions are more meaningful and superior to those generated by typical LSTMs and CNNs.

Third, we further extend the work by using an additional CNN layer to incorporate the structured local context together with the past and the future contexts attained by Bi-directional LSTM. A pooling scheme namely *Attention Pooling* is also used to enhance the information extraction capability of the pooling layer. Consequently, it is able to generate contextually superior captions.

Fourth, existing image captioning techniques use human-annotated real images for training and testing, which involve an expensive and time-consuming process. Moreover, nowadays bulk of the images are synthetic or generated by machines. There is also a need for generating captions for such images. We investigate the use of synthetic images for training and testing image captioning. We show that such images can help improving the captions of real images and they can effectively be used in caption generation of synthetic images.

PUBLICATIONS

Refereed Journal Articles

- Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga, (2019), “A Comprehensive Survey of Deep Learning for Image Captioning," ACM Computing Surveys (CSUR) 51, 6(2019), 118.
- Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammed Bennamoun (2020), “Text to Image Synthesis for Improved Image Captioning,” IEEE Access (Revision in preparation).
- Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammed Bennamoun (2020), “Image Captioning Leveraging Historical, Future, and Local Context,” IEEE Transactions on Multimedia (Under review).

Refereed International Conference Papers

- Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammad Bennamoun (2019), “Attention-based Image Captioning Using DenseNet Features," International Conference on Neural Information Processing (ICONIP), Sydney, Australia, pp. 109-117.
- Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammed Bennamoun (2019), “Bi-SAN-CAP: Bi-Directional Self-Attention for Image Captioning,” Digital Image Computing: Techniques and Applications (DICTA), Perth, Australia.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Everyday we encounter images in many ways; e.g., the Internet, news articles, document diagrams and advertisements. Humans usually find it easy to interpret these images and give a textual description. However, if machines need to give a textual description of an image, the machines need to understand the semantic and the context of the image. A long-standing goal in the field of Artificial Intelligence is to enable machines to see and understand the images of our surrounding [1].

1.1 Overview of Image Captioning

Image caption generation is the task of automatically generating a description of an image. It involves the understanding of the semantic of the image, which requires the understanding of the main objects, their various attributes, poses, and their interactions in an image. It also needs to infer the underlying semantic meanings to generate meaningful captions [2]. Figure 1.1 shows a few images with their captions. The captions “A couple of kids walking around with colourful umbrellas", “A green bird standing on peeled bananas in a background", and “A man in a soccer uniform playing soccer on a field" are the captions for the images in Figures 1.1(a), (b), and (c), respectively.

Image captioning is important for many reasons. For example, automatic image captioning can be useful for assisting visually impaired people, intelligent human computer interactions, and developing image search engines. Platforms such as Facebook and Twitter can directly generate description from the image, where we are (beach, cafe), what we wear and importantly what are we doing there [3], [4]. It can also be used for event summarization. Some examples of applications of image captioning are given in Figure 1.2, which shows captioning can be useful in (a) scene description for visually impaired people, (b) human-robot interaction, and (c) text-based image retrieval.

Image captioning is an important research area. Automatic generation of image captions requires both image understanding and a language description for that image. Image understanding is a core problem of Computer Vision. Language description is a part of Natural Language Understanding (NLU) [5]. A typical image captioning framework consists of an image encoder to learn features from an image and a language decoder to generate captions for that image.

(a) A couple of kids walking around with colourful umbrellas.

(b) A green bird standing on peeled bananas in a background.

(c) A man in a soccer uniform playing soccer on a field.

Figure 1.1: Examples of a few images with sample captions.

1.1.1 Image Understanding

Computer vision is the ability of machines to see and understand what is in their surroundings. It has all the methods to extract required information from the images. A significant amount of research is carried out in computer vision; especially visual recognition and visual understanding. Visual recognition involves identifying, localizing, and classifying objects of an image. Visual understanding requires object recognition as well extracting the complete detail of individual object and their associated relationship. Figure 1.3 shows a few examples of image understanding. Figure 1.3(a) has three main objects such as Person, Dog, and Chair, and Figure 1.3(b) contains different types of fruits such as Orange, Lemon, Grapes, Pear, and Lime. An image captioning method needs to correctly recognize multiple objects.

Features are important properties of an object. An object can have multiple features rather than only one attribute. For example, colours, contour lines, geometric lines or edges (gradient of pixel intensities) are popular choices [6].

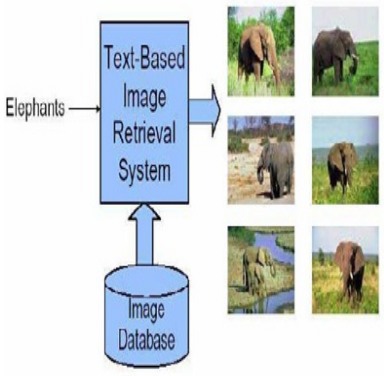
Features can be predefined (so called hand-crafted) or they can be learned. Hand-crafted features include Local Binary Pattern (LBP) [7], Histogram of Oriented Gradients(HOG) [8], Scale-invariant Feature Transform (SIFT) [9], and a combination of them. In these techniques, features are extracted from input data. However, real world image data are complex, redundant, and highly variable. The appearance of an object can be changed from image to image. Hand-crafted features are usually not robust and are computationally intensive. Therefore, extraction of hand-crafted features from a large and complex set of images is not feasible.

In deep learning based techniques, features are automatically learned. Convolutional Neural Networks (CNNs) are deep neural network architectures designed for working on images, videos, sound spectrograms in speech processing, character sequences in text and so on [10], [11]. They have made tasks much easier than hand crafted features-based techniques. CNNs have become capable to distinguish visual categories with some good levels of accuracy. These advancements are now widely used for face detection and recognition, personal photo search, perception in robotics, self-driving car and so on [12].

A convolutional neural network consists of one or more convolutional layers. These layers are then

(a) Scene description for visually impaired people.

(b) Human-robot interaction.



(c) Text-based image retrieval.

Figure 1.2: Applications of image captioning.

followed by one or more fully connected layers [13]. In this architecture, the lower layer is divided into some small regions called receptive fields. Each connection from the lower layer to the upper layer has a special value called weight. Each receptive field is mapped with the neuron of upper layers to extract features. Most popular Convolutional Neural architectures are described below:

LeCun Yann [10] developed the first architecture of Convolutional Neural Networks in 1990's. It is called LeNet. The LeNet architecture was mainly used to recognize zip codes, digits.

AlexNet [14] was developed by Alex Krizhevsky, IIya Sutskever and Geoff Hinton in 2012. The architecture of this network is very similar to LeNet. However it was deeper and bigger than LeNet. AlexNet contained total eight layers. First five were fully convolutional layers followed by fully connected layers.

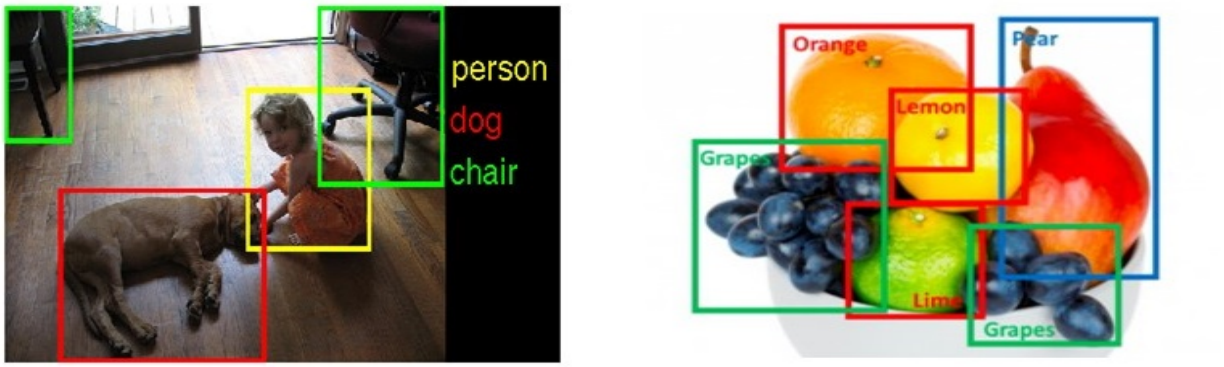
GoogleNet [15] was developed by Szegedy *et al.*. The main contribution was the addition of inception module. This module helps to reduce the number of parameters in the network.

Karen Simonyan and Andrew Zisserman developed the VGGNet [13]. The depth of the network is the main component for better performance. It has 16 convolutional layers and 3 fully connected layers. It performs 3×3 convolutions and 2×2 pooling from the beginning to the end.

He *et al.* [16] developed ResNet. It features special skip connections and a heavy use of batch normalization. This network is also missing fully connected layers at the end of the network.

In DenseNet [17], each layer has connections with every other layer in the network in a feed-forward manner. Therefore, L layers of DenseNet have $L(L+1)/2$ direct connections. As a result, the feature-maps of all preceding layers are used as inputs of the current layer, and its own feature-maps are used as inputs too all subsequent layers.

Pooling: Pooling is used to preserve more task-related information, more compact representations, and better robustness to noise and clutter [18]. They alleviate the problem of over-fitting. Then an activation function is used to produce a non-linear decision boundary from linear combinations of the weighted input [19]. A number of pooling functions such as max pooling [20], average pooling [21], and k-max pooling [11] are commonly used at pooling stage.



(a) An image of multiple objects: Person, Dog, and Chair.

(b) An image of different type of fruits: Orange, Lemon, Grapes, Pear, and Lime

Figure 1.3: Image Understanding.

1.1.2 Natural Language Understanding

According to the NLU point of view, generating text involves a series of steps. First, we have to know the aspects of the input which is called content selection and then we need to organize the content that is text planning and finally we need to verbalize it which is called surface realization. Surface realization requires lexicalization that means to select the right words, referential expression generation using appropriate pronouns, and then combining related information termed as aggregation [22].

Recurrent Neural Network (RNN) [23] and Long Short-Term Memory (LSTM) [24] are two popular deep learning-based language models that have shown great performances in many natural language understanding tasks including image captioning [25], [26], [27], [28], [29]. In Image captioning, image features extracted from a CNN encoder are given as input to RNN/LSTM. The RNN/LSTM then predicts the probability of each word given the previous words.

LSTM networks are a type of RNN that has special units in addition to standard units. LSTM units are able to actively maintain self-connecting loops involving an additional memory output. Thus they can maintain information in memory for long periods of time.

Another network, Gated Recurrent Unit (GRU) [30] has a similar structure to LSTM but it does not use separate memory cells and uses fewer gates to control the flow of information.

Bi-directional LSTM (BLSTM) [31] compute information in two ways: forward and backward directions. They combine the information using two hidden states and can preserve both past and future contexts.

CNNs can learn the internal hierarchical structure of the sentences and they are faster in processing than LSTMs. Therefore, recently, convolutional architectures are used in other sequence to sequence tasks, e.g., conditional image generation [32] and machine translation [33], [34], [35].

Attention: The attention mechanism [35], [36] is one of the most valuable breakthroughs in deep learning research in the last decade. It has particularly emerged as an improvement over encoder decoder-based NLU tasks [37], [38], [39]. Attention mechanisms such as soft attention [29], hard attention [29] have also been used in image captioning methods [40], [29], [41]. In these methods, attention mechanisms can dynamically focus on the relevant parts of the image while the output

(a) Caption 1: Football.
Caption 2: Red Football.

(b) Caption 1: Book and hand.
Caption 2: Book in hand.

(c) Caption 1: Car.
Caption 2: Tilted car.

Figure 1.4: Semantic Understanding.

sequences are being produced.

1.2 Main Research Challenges

Deep learning-based techniques, specifically CNNs have contributed substantially in understanding of an image. However, correct and precise recognition of objects contained in an image is one of the important requirements of image understanding. Despite wide research in this area, correct and precise recognition of multiple objects is still a challenging problem [42].

Most existing image captioning methods including deep learning-based techniques focus only on factual description of an image. During feature learning, these methods compress the entire scene into a fixed vector representation. As a result, they often lose the information of relevant objects in the scene [28],[29].

Image captioning is still a very challenging task because it requires not only to understand the objects and attributes but also to infer the underlying semantic information [43]. Figure 1.4 shows a couple of examples of semantic understanding. “Red Football" is semantically more meaningful than only “Footbal" in Figure 1.4a. Similarly, “Book in hand", and “Tilted car" are semantically correct and meaningful for the Figures 1.4b and 1.4c, respectively. The context of the relationship between objects of an image plays a significant role in semantic understanding. A suitable context (e.g., past, future) estimation can reduce the semantic gap between visual appearance and appropriate textual description of the image [44].

Existing image captioning techniques use human-annotated real images for training and testing, which involve an expensive and time-consuming process. Moreover, Nowadays a lot contents including images are generated automatically, e.g., for news, illustration, artwork, promotion, as well as for human computer interaction and augmented reality. There is a need to use these generated/synthetic images for training and testing image captioning methods. There is also a need to generate captions for such images.

1.3 Thesis Aims and Objectives

Given the Overview and the Research Challenges of Sections 1.1 and 1.2, we have the following aims and objectives in this thesis:

- To generate high quality captions of an image that can incorporate correct and relevant object information.
- Empowering deep networks with additional tools e.g., DenseNet with attention to generate meaningful and superior image captions.
- To incorporate past, future, and local contexts for generating semantically rich image captions.
- To demonstrate the usefulness of synthetic images for generating captions for both real and synthetic images.

1.4 Thesis Contributions

A Convolutional Neural Network (CNN) [10] is used as an image encoder to extract the visual representations of an input image and a Long Short-Term Memory (LSTM) [24] network is used as a language decoder to generate a caption for that image. A variety of CNNs and LSTM networks significantly contributed to the advancements of image captioning. In particular, CNN-LSTM based frameworks including attention mechanism have popularly been investigated for image captioning. In this thesis, we focus on advanced image captioning techniques. As such we have made four contributions in this thesis.

- DenseNet is a newer type of CNN where, each layer has connections with every other layer in the network in a feed-forward manner. The network reuses the feature-maps and uses concatenation for various operations instead of addition. Therefore, it can reduce the number of parameters and it can be memory efficient. Moreover, since each layer of DenseNet receives feature maps from all previous layers, it gets diversified features and tends to have rich patterns. For this reason, we use DenseNet for extracting features from images.

  Attention mechanisms [45] can focus on the parts of the image that are relevant, similar to the human visual system. Simultaneously, they can discard irrelevant information.

  Therefore, we investigate an attention based LSTM on image features extracted by DenseNet, which is a newer type of CNN. We show that this DenseNet features with attention mechanism can generate more relevant image captions than other CNNs.

- Typical LSTMs work only in one direction, forward direction. They can only preserve the past context using the hidden state because they have only seen the information from the past [46].

  Bi-directional self-attention computes attention both in forward and backward directions to encode the sequential information and feature-level information to handle the variation of contexts around the same word. It applies the forward positional mask to half of the sequence and the

backward positional mask to the remaining half. Consequently, it obtains diverse context of the words.

Therefore, we use bi-directional self-attention as a language decoder. Bi-directional decoder can capture the context in both forward and backward directions, i.e., the past context as well as any future context, in caption generation. Consequently, the generated captions are more meaningful and superior to those generated by typical LSTMs and CNNs.

- LSTMs have limitations in extracting the underlying hierarchical structure of a sequence [46]. Therefore, they do not perform well in capturing the local context of the sequence. In addition, typical LSTMs work only in forward direction. They can only preserve the past context using the hidden state because they have only seen the information from the past.

  BLSTM compute information in two ways: forward and backward directions. They combine the information using two hidden states and can preserve both past and future contexts. CNNs are also used in sequence modelling [32], [34]. CNNs can learn the internal hierarchical structures of sentences. They can independently capture local information contained in every word of a sentence. However, CNNs focus only on the local dependency of a sentence and do not perform well on a long expression. All the existing pooling functions such as max pooling [20], average pooling [21] have a tendency to discard context information to some extent.

  Therefore, we further extend our previous work by combining a BLSTM with a convolutional layer to extract comprehensive information, namely the past, the future, and the local context information of a caption. A pooling scheme namely *Attention Pooling* is also used to enhance the information extraction capability of the pooling layer. Consequently, it is able to generate contextually superior captions.

- To the best of our knowledge, there is no method available in image captioning which use synthetic images. Existing image caption generators are only trained on labelled real images. It is important to develop caption generators for synthetic images as well.

  Therefore, we investigate and analyse the use of synthetic images for training and testing the image captioning methods. To achieve the goal, we propose a pipeline composed of a GAN Module to generate synthetic images and an image captioning module to generate captions. we demonstrate that such images can help improving the captions of real images and they can effectively be used in caption generation of synthetic images.

1.5 Thesis Outline

The remainder of this thesis is structured as follows. **Chapter 2** gives a comprehensive review of the state-of-the-art deep learning based methods of image captioning. This chapter also includes the details of all the datasets and evaluation metrics used for image captioning. **Chapter 3** deals with the generation of image captions addressing correct and relevant object information. The following chapter **(Chapter 4)** tackles the limitations of LSTMs and CNNs in using as language decoder. In **Chapter 4**, we focus on the importance of the past and the future context for generating superior

and rich semantic image captions. **Chapter 5** extends **Chapter 4** by adding local context together with the past and the future contexts to generate more meaningful and accurate captions. Concerned with data augmentation, **Chapter 6** explores the effectiveness of using synthetic images for image captioning to further improve the quality of the generated captions. Finally, the conclusions and ideas for future work are presented in **Chapter 7**.

In **Chapter 2**, we provide a comprehensive literature review of image captioning explaining different methods, datasets, and evaluation metrics. In this chapter, we group the methods of image captioning into different categories. The categories are discussed briefly with a detailed discussion of deep learning based methods. Image captioning methods use a number of publicly available datasets and evaluation metrics to demonstrate the empirical results. The common and popular datasets and evaluation metrics used in these methods are also discussed in this chapter.

In **Chapter 3**, we develop a method with an attention based LSTM network and a DenseNet. DenseNet is capable to extract robust features from an image and attention based LSTM can selectively focus on the relevant features of an image. We show that the combination of DenseNet and LSTM with attention mechansim can generate image captions with correct and relevant object information.

In **Chapter 4**, we specifically address the effectiveness of using both past and future contexts. We introduce a Bi-directional self-attention for image captioning to incorporate the past and the future contexts in generating rich semantic image captions. Bi-directional self-attention can work in both forward and backward directions to capture the past and the future contexts information.

In **Chapter 5**, we propose an architecture where a BLSTM is combined with a convolutional layer in language decoder. The architecture also includes an *Attention Pooling* mechanism that can retain most significant information at the pooling stage. This combined architecture of language decoder with the attention pooling technique can generate image caption with comprehensive information namely past, future, and local contexts.

In **Chapter 6**, we explore the ideas of using synthetic images for image captioning as data augmentation. The content of this chapter demonstrates the analysis and the effectiveness of using synthetic images for improving the quality of the generated captions of real images. It also includes a discussion and analysis of the importance of generating captions for synthetic images.

Finally, we provide conclusions and future work in the last chapter (**Chapter 7**).

Chapter 2

Literature Review

ABSTRACT

Generating a description of an image is called image captioning. Image captioning requires to recognize the important objects, their attributes and their relationships in an image. It also needs to generate syntactically and semantically correct sentences. Deep learning-based techniques are capable of handling the complexities and challenges of image captioning. In this survey paper, we aim to present a comprehensive review of existing deep learning-based image captioning techniques. We discuss the foundation of the techniques to analyze their performances, strengths and limitations. We also discuss the datasets and the evaluation metrics popularly used in deep learning based automatic image captioning.

2.1 Introduction

Every day, we encounter a large number of images from various sources such as the internet, news articles, document diagrams and advertisements. These sources contain images that viewers would have to interpret themselves. Most images do not have a description, but the human can largely understand them without their detailed captions. However, machine needs to interpret some form of image captions if humans need automatic image captions from it.

Image captioning is important for many reasons. For example, they can be used for automatic image indexing. Image indexing is important for Content-Based Image Retrieval (CBIR) and therefore, it can be applied to many areas, including biomedicine, commerce, the military, education, digital libraries, and web searching. Social media platforms such as Facebook and Twitter can directly generate descriptions from images. The descriptions can include where we are (e.g., beach, cafe), what we wear and importantly what we are doing there.

Image captioning is a popular research area of Artificial Intelligence (AI) that deals with image understanding and a language description for that image. Image understanding needs to detect and

This chapter is published in the journal of ACM Computing Surveys (CSUR) 51, 6(2019), 118, under the title of “A Comprehensive Survey of Deep Learning for Image Captioning".

recognize objects. It also needs to understand scene type or location, object properties and their interactions. Generating well-formed sentences requires both syntactic and semantic understanding of the language [47].

Understanding an image largely depends on obtaining image features. The techniques used for this purpose can be broadly divided into two categories: (1) Traditional machine learning based techniques and (2) Deep machine learning based techniques.

In traditional machine learning, hand crafted features such as Local Binary Patterns (LBP) [7], Scale-Invariant Feature Transform (SIFT) [9], the Histogram of Oriented Gradients (HOG) [8], and a combination of such features are widely used. In these techniques, features are extracted from input data. They are then passed to a classifier such as Support Vector Machines (SVM) [48] in order to classify an object. Since hand crafted features are task specific, extracting features from a large and diverse set of data is not feasible. Moreover, real world data such as images and video are complex and have different semantic interpretations.

On the other hand, in deep machine learning based techniques, features are learned automatically from training data and they can handle a large and diverse set of images and videos. For example, Convolutional Neural Networks (CNN) [10] are widely used for feature learning, and a classifier such as Softmax is used for classification. CNN is generally followed by Recurrent Neural Networks (RNN) in order to generate captions.

In the last 5 years, a large number of articles have been published on image captioning with deep machine learning being popularly used. Deep learning algorithms can handle complexities and challenges of image captioning quite well. So far, only three survey papers [5, 49, 50] have been published on this research topic. Although the papers have presented a good literature survey of image captioning, they could only cover a few papers on deep learning because the bulk of them was published after the survey papers. These survey papers mainly discussed template based, retrieval based, and a very few deep learning-based novel image caption generating models. However, a large number of works have been done on deep learning-based image captioning. Moreover, the availability of large and new datasets has made the learning-based image captioning an interesting research area. To provide an abridged version of the literature, we present a survey mainly focusing on the deep learning-based papers on image captioning.

The main aim of this paper is to provide a comprehensive survey of deep learning for image captioning. First, we group the existing image captioning articles into three main categories: (1) Template-based Image captioning, (2) Retrieval-based image captioning, and (3) Novel image caption generation. The categories are discussed briefly in Section 2.2. Most deep learning based image captioning methods fall into the category of novel caption generation. Therefore, we focus only on novel caption generation with deep learning. Second, we group the deep learning-based image captioning methods into different categories namely (1) Visual space-based, (2) Multimodal space-based, (3) Supervised learning, (4) Other deep learning, (5) Dense captioning, (6) Whole scene-based, (7) Encoder-Decoder Architecture-based, (8) Compositional Architecture-based, (9) LSTM (Long Short-Term Memory) [24] language model-based, (10) Others language model-based, (11) Attention-Based, (12) Semantic concept-based, (13) Stylized captions, and (12) Novel object-based image captioning. We discuss all the categories

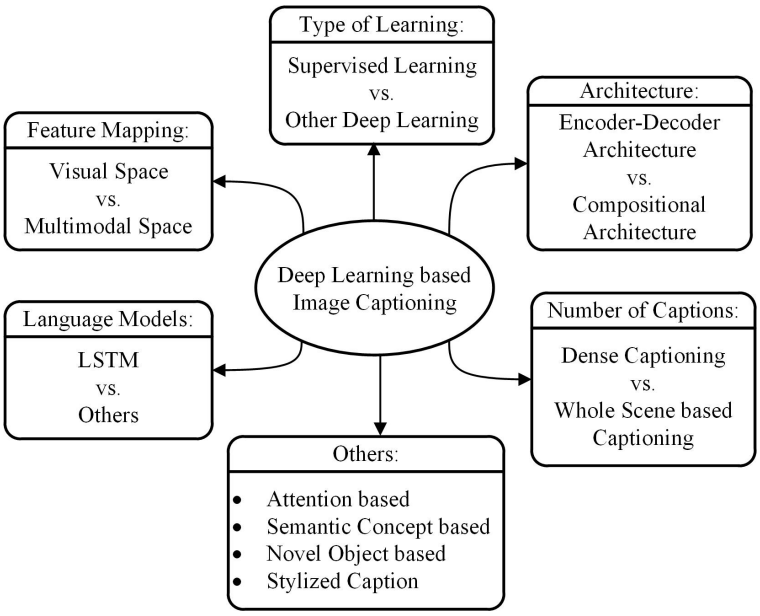


Figure 2.1: An overall taxonomy of deep learning-based image captioning.

in Section 2.3. We provide an overview of the datasets and commonly used evaluation metrics for measuring the quality of image captions in Section 2.4. We also discuss and compare the results of different methods in Section 2.5. Finally, we give a brief discussion and future research directions in Section 2.6 and then a conclusion in Section 2.7.

2.2 Image Captioning Methods

In this section, we review and describe the main categories of existing image captioning methods and they include template-based image captioning, retrieval-based image captioning, and novel caption generation.

Template-based approaches have fixed templates with a number of blank slots to generate captions. In these approaches, different objects, attributes, actions are detected first and then the blank spaces in the templates are filled. For example, Farhadi et al. [51] use a triplet of scene elements to fill the template slots for generating image captions. Li et al. [52] extract the phrases related to detected objects, attributes and their relationships for this purpose. A Conditional Random Field (CRF) is adopted by Kulkarni et al. [53] to infer the objects, attributes, and prepositions before filling in the gaps. Template-based methods can generate grammatically correct captions. However, templates are predefined and cannot generate variable-length captions. Moreover, later on, parsing based language models have been introduced in image captioning [54, 55, 56, 57, 58] which are more powerful than fixed template-based methods. Therefore, in this paper, we do not focus on these template based methods.

Captions can be retrieved from visual space and multimodal space. In retrieval-based approaches, captions are retrieved from a set of existing captions. Retrieval based methods first find the visually similar images with their captions from the training data set. These captions are called candidate captions. The captions for the query image are selected from these captions pool [59, 60, 61, 62]. These

methods produce general and syntactically correct captions. However, they cannot generate image specific and semantically correct captions.

Novel captions can be generated from both visual space and multimodal space. A general approach of this category is to analyze the visual content of the image first and then generate image captions from the visual content using a language model [63, 29, 64, 65]. These methods can generate new captions for each image that are semantically more accurate than previous approaches. Most novel caption generation methods use deep machine learning based techniques. Therefore, deep learning based novel image caption generating methods are our main focus in this literature.

An overall taxonomy of deep learning-based image captioning methods is depicted in Figure 2.1. The figure illustrates the comparisons of different categories of image captioning methods. Novel caption generation-based image caption methods mostly use visual space and deep machine learning based techniques. Captions can also be generated from multimodal space. Deep learning-based image captioning methods can also be categorized on learning techniques: Supervised learning, Reinforcement learning, and Unsupervised learning. We group the reinforcement learning and unsupervised learning into Other Deep Learning. Usually captions are generated for a whole scene in the image. However, captions can also be generated for different regions of an image (Dense captioning). Image captioning methods can use either simple Encoder-Decoder architecture or Compositional architecture. There are methods that use attention mechanism, semantic concept, and different styles in image descriptions. Some methods can also generate description for unseen objects. We group them into one category as “Others". Most of the image captioning methods use LSTM as language model. However, there are a number of methods that use other language models such as CNN and RNN. Therefore, we include a language model-based category as “LSTM vs. Others".

2.3 Deep Learning Based Image Captioning Methods

We draw an overall taxonomy in Figure 2.1 for deep learning-based image captioning methods. We discuss their similarities and dissimilarities by grouping them into visual space vs. multimodal space, dense captioning vs. captions for the whole scene, Supervised learning vs. Other deep learning, Encoder-Decoder architecture vs. Compositional architecture, and one ‘Others’ group that contains Attention-Based, Semantic Concept-Based, Stylized captions, and Novel Object-Based captioning. We also create a category named LSTM vs. Others.

A brief overview of the deep learning-based image captioning methods is shown in Table 2.1. Table 2.1 contains the name of the image captioning methods, the type of deep neural networks used to encode image information, and the language models used in describing the information. In the final column, we give a category label to each captioning technique based on the taxonomy in Figure 2.1.

2.3.1 Visual Space vs. Multimodal Space

Deep learning-based image captioning methods can generate captions from both visual space and multimodal space. Understandably image captioning datasets have the corresponding captions as text. In the visual space-based methods, the image features and the corresponding captions are independently

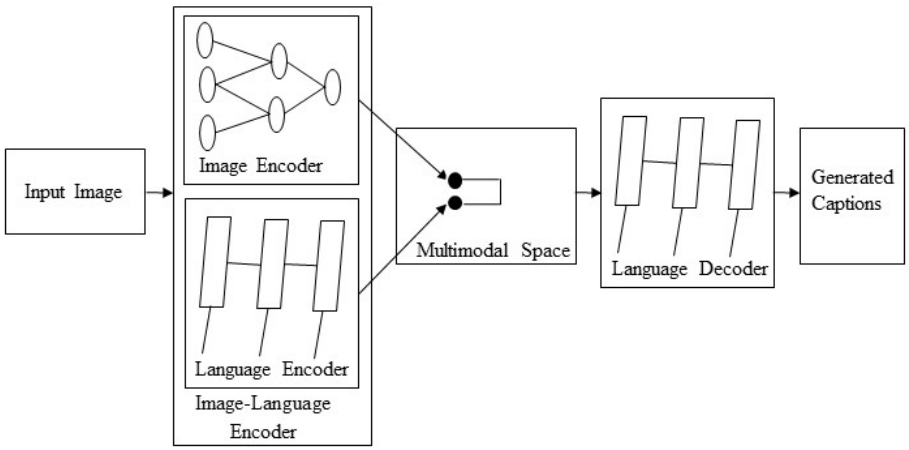


Figure 2.2: A block diagram of multimodal space-based image captioning.

passed to the language decoder. In contrast, in a multimodal space case, a shared multimodal space is learned from the images and the corresponding caption-text. This multimodal representation is then passed to the language decoder.

Visual Space

Bulk of the image captioning methods use visual space for generating captions. These methods are discussed in Section 2.3.2 to Section 2.3.5.

Multimodal Space

The architecture of a typical multimodal space-based method contains a language Encoder part, a vision part, a multimodal space part, and a language decoder part. A general diagram of multimodal space-based image captioning methods is shown in Figure 2.2. The vision part uses a deep convolutional neural network as a feature extractor to extract the image features. The language encoder part extracts the word features and learns a dense feature embedding for each word. It then forwards the semantic temporal context to the recurrent layers. The multimodal space part maps the image features into a common space with the word features. The resulting map is then passed to the language decoder which generates captions by decoding the map.

The methods in this category follow the following steps:

1. Deep neural networks and multimodal neural language model are used to learn both image and text jointly in a multimodal space.

2. The language generation part generates captions using the information from Step 1 .

An initial work in this area proposed by Kiros et al. [66]. The method applies a CNN for extracting image features in generating image captions. It uses a multimodal space that represents both image and text jointly for multimodal representation learning and image caption generation. It also introduces the multimodal neural language models such as Modality-Biased Log-Bilinear Model (MLBL-B) and the Factored 3-way Log-Bilinear Model (MLBL-F) of [103] followed by AlexNet [14]. Unlike most previous

Reference	Image Encoder	Language Model	Category
Kiros et al. 2014 [66]	AlexNet	LBL	MS,SL,WS,EDA
Kiros et al. 2014 [63]	AlexNet, VGGNet	1. LSTM 2. SC-NLM	MS,SL,WS,EDA
Mao et al. 2014 [67]	AlexNet	RNN	MS,SL,WS
Karpathy et al. 2014 [68]	AlexNet	DTR	MS,SL,WS,EDA
Mao et al. 2015 [69]	AlexNet, VGGNet	RNN	MS,SL,WS
Chen et al. 2015 [70]	VGGNet	RNN	VS,SL,WS,EDA
Fang et al. 2015 [71]	AlexNet, VGGNet	MELM	VS,SL,WS,CA
Jia et al. 2015 [72]	VGGNet	LSTM	VS,SL,WS,EDA
Karpathy et al. 2015 [1]	VGGNet	RNN	MS,SL,WS,EDA
Vinyals et al. 2015 [28]	GoogLeNet	LSTM	VS,SL,WS,EDA
Xu et al. 2015 [29]	AlexNet	LSTM	VS,SL,WS,EDA,AB
Jin et al. 2015 [73]	VGGNet	LSTM	VS,SL,WS,EDA,AB
Wu et al. 2016 [74]	VGGNet	LSTM	VS,SL,WS,EDA,AB
Sugano et at. 2016 [75]	VGGNet	LSTM	VS,SL,WS,EDA,AB
Mathews et al. 2016 [76]	GoogLeNet	LSTM	VS,SL,WS,EDA,SC
Wang et al. 2016 [2]	AlexNet, VGGNet	LSTM	VS,SL,WS,EDA
Johnson et al. 2016 [77]	VGGNet	LSTM	VS,SL,DC,EDA
Mao et al. 2016 [78]	VGGNet	LSTM	VS,SL,WS,EDA
Wang et al. 2016 [79]	VGGNet	LSTM	VS,SL,WS,CA
Tran et al. 2016 [80]	ResNet	MELM	VS,SL,WS,CA
Ma et al. 2016 [81]	AlexNet	LSTM	VS,SL,WS,CA
You et al. 2016 [65]	GoogLeNet	RNN	VS,SL,WS,EDA,SCB
Yang et al. 2016 [82]	VGGNet	LSTM	VS,SL,DC,EDA
Anne et al. 2016 [83]	VGGNet	LSTM	VS,SL,WS,CA,NOB
Yao et al. 2017 [64]	GoogLeNet	LSTM	VS,SL,WS,EDA,SCB
Lu et al. 2017 [84]	ResNet	LSTM	VS,SL,WS,EDA,AB
Chen et al. 2017 [85]	VGGNet, ResNet	LSTM	VS,SL,WS,EDA,AB
Gan et al. 2017 [86]	ResNet	LSTM	VS,SL,WS,CA,SCB
Pedersoli et al. 2017 [87]	VGGNet	RNN	VS,SL,WS,EDA,AB
Ren et al. 2017 [88]	VGGNet	LSTM	VS,ODL,WS,EDA
Park et al. 2017 [89]	ResNet	LSTM	VS,SL,WS,EDA,AB
Wang et al. 2017 [90]	ResNet	LSTM	VS,SL,WS,EDA
Tavakoli et al. 2017 [42]	VGGNet	LSTM	VS,SL,WS,EDA,AB
Liu et al. 2017 [91]	VGGNet	LSTM	VS,SL,WS,EDA,AB
Gan et al. 2017 [92]	ResNet	LSTM	VS,SL,WS,EDA,SC
Dai et al. 2017 [93]	VGGNet	LSTM	VS,ODL,WS,EDA
Shetty et al. 2017 [94]	GoogLeNet	LSTM	VS,ODL,WS,EDA
Liu et al. 2017 [95]	Inception-V3	LSTM	VS,ODL,WS,EDA
Gu et al. 2017 [96]	VGGNet	1. Language CNN 2. LSTM	VS,SL,WS,EDA
Yao et al. 2017 [97]	VGGNet	LSTM	VS,SL,WS,CA,NOB
Rennie et al. 2017 [98]	ResNet	LSTM	VS,ODL,WS,EDA
Vsub et al. 2017 [99]	VGGNet	LSTM	VS,SL,WS,CA,NOB
Zhang et al. 2017 [100]	Inception-V3	LSTM	VS,ODL,WS,EDA
Wu et al. 2018 [101]	VGGNet	LSTM	VS,SL,WS,EDA,SCB
Aneja et al. 2018 [46]	VGGNet	Language CNN	VS,SL,WS,EDA
Wang et al. 2018 [102]	VGGNet	Language CNN	VS,SL,WS,EDA

Table 2.1: An overview of the deep learning-based approaches for image captioning (VS=Visual Space, MS=Multimodal Space, SL=Supervised Learning, ODL=Other Deep Learning, DC=Dense Captioning, WS=Whole Scene, EDA=Encoder-Decoder Architecture, CA=Compositional Architecture, AB=Attention-Based, SCB=Semantic Concept-Based, NOB=Novel Object-Based, SC=Stylized Caption).

approaches, this method does not rely on any additional templates, structures, or constraints. Instead it depends on the high level image features and word representations learned from deep neural networks and multimodal neural language models respectively. The neural language models have limitations to handle a large amount of data and are inefficient to work with long term memory [104].

Kiros et al. [66] extended their work in [63] to learn a joint image sentence embedding where LSTM is used for sentence encoding and a new neural language model called the structure-content neural language model (SC-NLM) is used for image captions generations. The SC-NLM has an advantage over existing methods in that it can extricate the structure of the sentence to its content produced by the encoder. It also helps them to achieve significant improvements in generating realistic image captions over the approach proposed by [66].

Karpathy et al. [68] proposed a deep, multimodal model, embedding of image and natural language data for the task of bidirectional images and sentences retrieval. The previous multimodal-based methods use a common, embedding space that directly maps images and sentences. However, this method works at a finer level and embeds fragments of images and fragments of sentences. This method breaks down the images into a number of objects and sentences into a dependency tree relations (DTR) [105] and reasons about their latent, inter-modal alignment. It shows that the method achieves significant improvements in the retrieval task compared to other previous methods. This method has a few limitations as well. In terms of modelling, the dependency tree can model relations easily but they are not always appropriate. For example, a single visual entity might be described by a single complex phrase that can be split into multiple sentence fragments. The phrase "black and white dog" can be formed into two relations (CONJ, black, white) and (AMOD, white, dog). Again, for many dependency relations we do not find any clear mapping in the image (For example: "each other" cannot be mapped to any object).

Mao et al. [69] proposed a multimodal Recurrent Neural Network (m-RNN) method for generating novel image captions. This method has two sub-networks: a deep recurrent neural network for sentences and a deep convolutional network for images. These two sub-networks interact with each other in a multimodal layer to form the whole m-RNN model. Both image and fragments of sentences are given as input in this method. It calculates the probabilty distribution to generate the next word of captions. There are five more layers in this model: Two-word embedding layers, a recurrent layer, a multimodal layer and a SoftMax layer.

Kiros et al. [66] proposed a method that is built on a Log-Bilinear model and used AlexNet to extract visual features. This multimodal recurrent neural network method is closely related to the method of Kiros et al. [66]. Kiros et al. use a fixed length context (i.e. five words), whereas in this method, the temporal context is stored in a recurrent architecture, which allows an arbitrary context length. The two word embedding layers use one hot vector to generate a dense word representation. It encodes both the syntactic and semantic meaning of the words. The semantically relevant words can be found by calculating the Euclidean distance between two dense word vectors in embedding layers. Most sentence-image multimodal methods [68, 106, 107, 63] use pre-computed word embedding vectors to initialize their model. In contrast, this method randomly initializes word embedding layers and learn them from the training data. This helps them to generate better image captions than the previous

methods. Many image captioning methods [67, 66, 68] are built on recurrent neural networks at the contemporary times. They use a recurrent layer for storing visual information. However, (m-RNN) use both image representations and sentence fragments to generate captions. It utilizes the capacity of the recurrent layer more efficiently that helps to achieve a better performance using a relatively small dimensional recurrent layer.

Chen et al. [70] proposed another multimodal space-based image captioning method. The method can generate novel captions from image and restore visual features from the given description. It also can describe a bidirectional mapping between images and their captions. Many of the existing methods [60, 107, 68] use joint embedding to generate image captions. However, they do not use reverse projection that can generate visual features from captions. On the other hand, this method dynamically updates the visual representations of the image from the generated words. It has an additional recurrent visual hidden layer with RNN that makes reverse projection.

2.3.2 Supervised Learning vs. Other Deep Learning

In supervised learning, training data come with desired output called *label*. Unsupervised learning, on the other hand, deals with unlabeled data. Generative Adversarial Networks (GANs) [108] are a type of unsupervised learning techniques. Reinforcement learning is another type of machine learning approach where the aims of an agent are to discover data and/or labels through exploration and a reward signal. A number of image captioning methods use reinforcement learning and GAN based approaches. These methods sit in the category of “Other Deep Learning".

Supervised Learning-Based Image Captioning

Supervised learning-based networks have successfully been used for many years in image classification [14, 16, 13, 15], object detection [109, 110, 111], and attribute learning [112]. This progress makes researchers interested in using them in automatic image captioning [28, 69, 1, 70]. In this paper, we have identified a large number of supervised learning-based image captioning methods. We classify them into different categories: (i) Encoder-Decoder Architecture, (ii) Compositional Architecture, (iii) Attention-based, (iv) Semantic concept-based, (v) Stylized captions, (vi) Novel object-based, and (vii) Dense image captioning.

Other Deep Learning-Based Image Captioning

In our day to day life, data are increasing with unlabled data because it is often impractical to accurately annotate data. Therefore, recently, researchers are focusing more on reinforcement learning and unsupervised learning-based techniques for image captioning.

A reinforcement learning approach is designed by a number of parameters such as agent, state, action, reward function, policy, and value. The agent chooses an action, receives reward values, and moves to a new state. policies are defined by actions and values are defined by reward function. The agent attempts to select the action with the expectation of having a maximum long-term reward. It needs continuous state and action information to provide the guarantees of a reward function. Traditional reinforcement learning approaches face a number of limitations such as the lack of guarantees of a

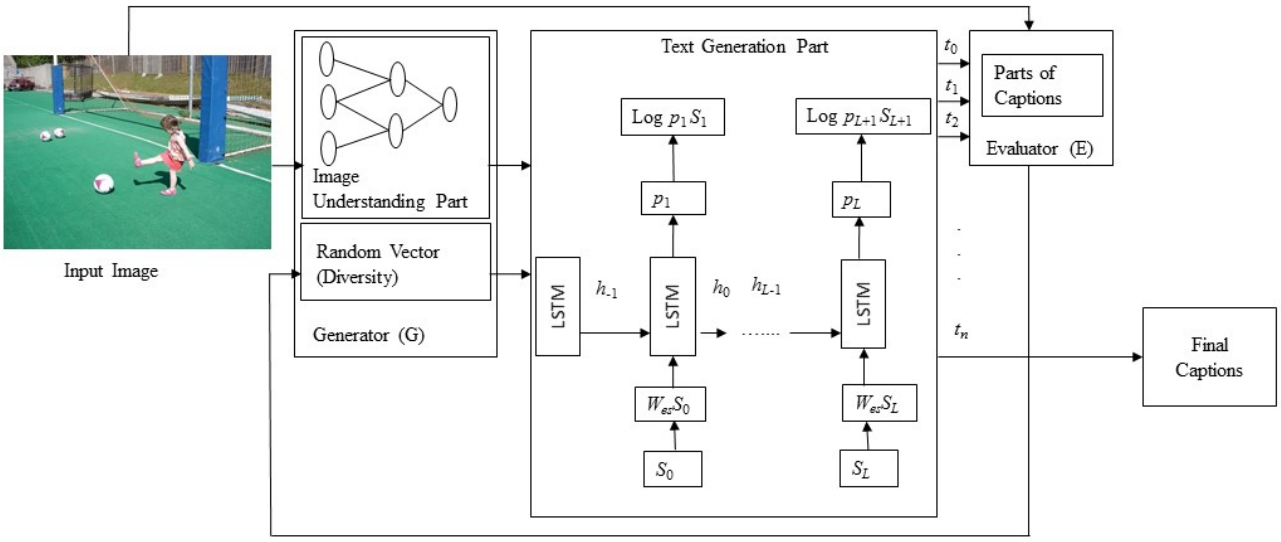


Figure 2.3: A block diagram of other deep learning-based captioning.

reward function and uncertain state-action information. Policy gradient methods [113] are a type of reinforcement learning that can choose a specific policy for a specific action using gradient descent and optimization techniques. The policy can incorporate domain knowledge for the action that guarantees convergence. Thus, policy gradient methods require fewer parameters than reward function based approaches.

Existing deep learning-based image captioning methods use variants of image encoders to extract image features. The features are then fed into the neural network-based language decoders to generate captions. The methods have two main issues: (i) They are trained using maximum likelihood estimation and back-propagation [114] approaches. In this case, the next word is predicted given the image and all the previously generated ground-truth words. Therefore, the generated captions look-like ground-truth captions. This phenomenon is called exposure bias [115] problem. (ii) Evaluation metrics at test time are non-differentiable. Ideally sequence models for image captioning should be trained to avoid exposure bias and directly optimise metrics for the test time. A typical architecture of reinforcement learning-based image captioning method has two networks: (i) the policy network and (ii) the value network. Sometimes they are referred to as actor and critic, respectively. The critic (value network) can be used in estimating the expected future reward to train the actor (captioning policy network). Reinforcement learning-based image captioning methods sample the next token from the model based on the rewards they receive in each state. Policy gradient methods in reinforcement learning can optimize the gradient in order to predict the cumulative long-term rewards. Therefore, it can solve the non-differentiable problem of evaluation metrics.

The methods in this category follow the following steps:

1. A CNN and RNN based combined network generates captions.
2. Another CNN-RNN based network evaluates the captions and send feedback to the first network to generate high quality captions.

A block diagram of a typical method of this category is shown in Figure 2.3.

Ren et al. 2017 [88] introduced a novel reinforcement learning based image captioning method. The architecture of this method has two networks that jointly compute the next best word at each time step. The "policy network" works as local guidance and helps to predict next word based on the current state. The "value network"' works as global guidance and evaluates the reward value considering all the possible extensions of the current state. This mechanism is able to adjust the networks in predicting the correct words. Therefore, it can generate good captions similar to ground truth captions at the end. It uses an actor-critic reinforcement learning model [116] to train the whole network. Visual semantic embedding [117, 118] is used to compute the actual reward value in predicting the correct word. It also helps to measure the similarity between images and sentences that can evaluate the correctness of generated captions.

Rennie et al. [98] proposed another reinforcement learning based image captioning method. The method utilizes the test-time inference algorithm to normalize the reward rather than estimating the reward signal and normalization in training time. It shows that this test-time decoding is highly effective for generating quality image captions.

Zhang et al. [100] proposed an actor-critic reinforcement learning-based image captioning method. The method can directly optimize non-differentiable problems of the existing evaluation metrics. The architecture of the actor-critic method consists of a policy network (actor) and a value network (critic). The actor treats the job as sequential decision problem and can predict the next token of the sequence. In each state of the sequence, the network will receive a task-specific reward (in this case, it is evaluation metrics score). The job of the critic is to predict the reward. If it can predict the expected reward, the actor will continue to sample outputs according to its probability distribution.

GAN based methods can learn deep features from unlabeled data. They achieve this representations applying a competitive process between a pair of networks: the Generator and the Discriminator. GANs have already been used successfully in a variety of applications, including image captioning[93, 94], image to image translation [119], text to image synthesis [120, 121], and text generation [122, 123].

There are two issues with GAN. First, GAN can work well in generating natural images from real images because GANs are proposed for real-valued data. However, text processing is based on discrete numbers. Therefore, such operations are non-differentiable, making it difficult to apply back-propagation directly. Policy gradients apply a parametric function to allow gradients to be back-propagated. Second, the evaluator faces problems in vanishing gradients and error propagation for sequence generation. It needs a probable future reward value for every partial description. Monte Carlo rollouts [124] is used to compute this future reward value.

GAN based image captioning methods can generate a diverse set of image captions in contrast to conventional deep convolutional network and deep recurrent network based model. Dai et al. [93] also proposed a GAN based image captioning method. However, they do not consider multiple captions for a single image. Shetty et al. [94] introduced a new GAN based image captioning method. This method can generate multiple captions for a single image and showed impressive improvements in generating diverse captions. GANs have limitations in backpropagating the discrete data. Gumbel sampler [125, 126] is used to overcome the discrete data problem. The two main parts of this adversarial network are the generator and the discriminator. During training, generator learns the loss value provided by the

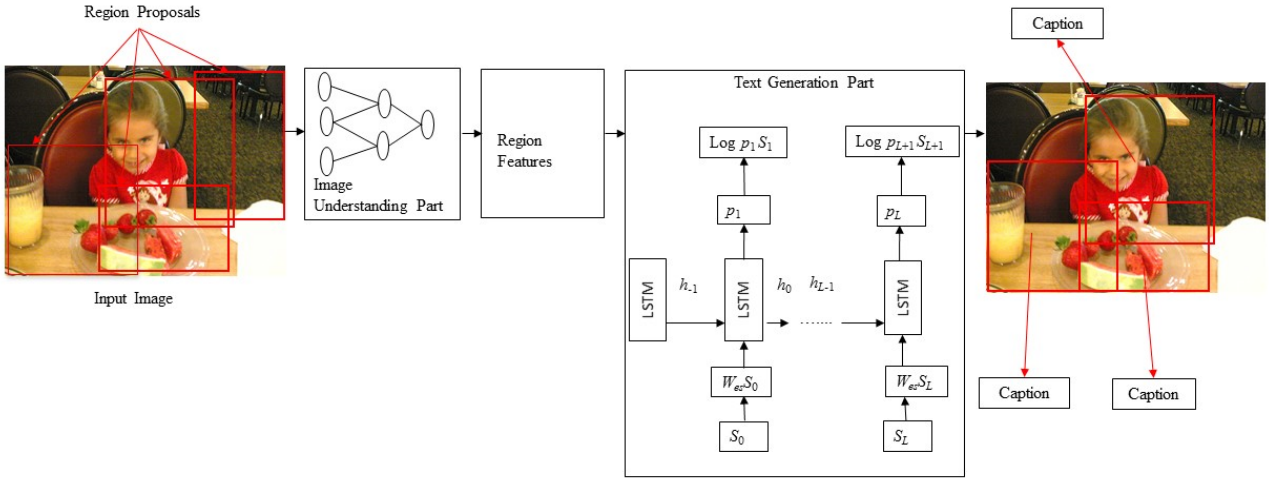


Figure 2.4: A block diagram of dense captioning.

discriminator instead of learning it from explicit sources. Discriminator has true data distribution and can discriminate between generator-generated samples and true data samples. This allows the network to learn diverse data distribution. Moreover, the network classifies the generated caption sets either real or fake. Thus, it can generate captions similar to human generated one.

2.3.3 Dense Captioning vs. Captions for the whole scene

In dense captioning, captions are generated for each region of the scene. Other methods generate captions for the whole scene.

Dense Captioning

The previous image captioning methods can generate only one caption for the whole image. They use different regions of the image to obtain information of various objects. However, these methods do not generate region wise captions.

Johnson et al. [77] proposed an image captioning method called DenseCap. This method localizes all the salient regions of an image and then it generates descriptions for those regions.

A typical method of this category has the following steps:

1. Region proposals are generated for the different regions of the given image.
2. CNN is used to obtain the region-based image features.
3. The outputs of Step 2 are used by a language model to generate captions for every region.

A block diagram of a typical dense captioning method is given in Figure 2.4.

Dense captioning [77] proposes a fully convolutional localization network architecture, which is composed of a convolutional network, a dense localization layer, and an LSTM [24] language model. The dense localization layer processes an image with a single, efficient forward pass, which implicitly predicts a set of region of interest in the image. Thereby, it requires no external region proposals unlike to Fast

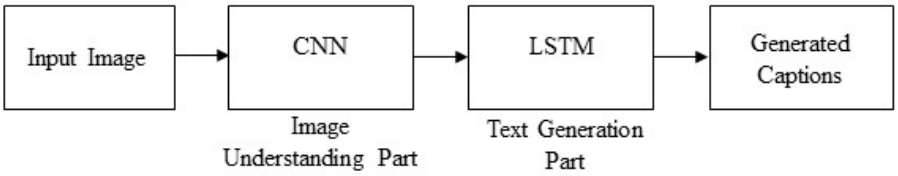


Figure 2.5: A block diagram of simple Encoder-Decoder architecture-based image captioning.

R-CNN or a full network (i.e., RPN (Region Proposal Network [109])) of Faster R-CNN. The working principle of the localization layer is related to the work of Faster R-CNN [111]. However, Johnson et al. [77] use a differential, spatial soft attention mechanism [127, 128] and bilinear interpolation [128] instead of ROI pooling mechanism [109]. This modification helps the method to backpropagate through the network and smoothly select the active regions. It uses Visual Genome [129] dataset for the experiments in generating region level image captions.

One description of the entire visual scene is quite subjective and is not enough to bring out the complete understanding. Region-based descriptions are more objective and detailed than global image description. The region-based description is known as dense captioning. There are some challenges in dense captioning. As regions are dense, one object may have multiple overlapping regions of interest. Moreover, it is very difficult to recognize each target region for all the visual concepts. Yang et al. [82] proposed another dense captioning method. This method can tackle these challenges. First, it addresses an inference mechanism that jointly depends on the visual features of the region and the predicted captions for that region. This allows the model to find an appropriate position of the bounding box. Second, they apply a context fusion that can combine context features with the visual features of respective regions to provide a rich semantic description.

Captions for the whole scene

Encoder-Decoder architecture, Compositional architecture, attention-based, semantic concept-based, stylized captions, Novel object-based image captioning, and other deep learning networks-based image captioning methods generate single or multiple captions for the whole scene.

2.3.4 Encoder-Decoder Architecture vs. Compositional Architecture

Some methods use just simple vanilla encoder and decoder to generate captions. However, other methods use multiple networks for it.

Encoder-Decoder Architecture-Based Image captioning

The neural network-based image captioning methods work as just simple end to end manner. These methods are very similar to the encoder-decoder framework-based neural machine translation [130]. In this network, global image features are extracted from the hidden activations of CNN and then fed them into an LSTM to generate a sequence of words.

A typical method of this category has the following general steps:

1. A vanilla CNN is used to obtain the scene type, to detect the objects and their relationships.

2. The output of Step 1 is used by a language model to convert them into words, combined phrases that produce an image captions.

A simple block diagram of this category is given in Figure 2.5.

Vinyals et al. [28] proposed a method called Neural Image Caption Generator (NIC). The method uses a CNN for image representations and an LSTM for generating image captions. This special CNN uses a novel method for batch normalization and the output of the last hidden layer of CNN is used as an input to the LSTM decoder. This LSTM is capable of keeping track of the objects that already have been described using text. NIC is trained based on maximum likelihood estimation.

In generating image captions, image information is included to the initial state of an LSTM. The next words are generated based on the current time step and the previous hidden state. This process continues until it gets the end token of the sentence. Since image information is fed only at the beginning of the process, it may face vanishing gradient problems. The role of the words generated at the beginning is also becoming weaker and weaker. Therefore, LSTM is still facing challenges in generating long length sentences [131, 132]. Therefore, Jia et al. [72] proposed an extension of LSTM called guided LSTM (gLSTM). This gLSTM can generate long sentences. In this architecture, it adds global semantic information to each gate and cell state of LSTM. It also considers different length normalization strategies to control the length of captions. Semantic information is extracted in different ways. First, it uses a cross-modal retrieval task for retrieving image captions and then semantic information is extracted from these captions. The semantic based information can also be extracted using a multimodal embedding space.

Mao et al. [78] proposed a special type of text generation method for images. This method can generate a description for an specific object or region that is called referring expression [133, 134, 135, 136, 137, 138, 139]. Using this expression it can then infer the object or region which is being described. Therefore, generated description or expression is quite unambiguous. In order to address the referring expression, this method uses a new dataset called ReferIt dataset [139] based on popular MS COCO dataset.

Previous CNN-RNN based image captioning methods use LSTM that are unidirectional and relatively shallow in depth. In unidirectional language generation techniques, the next word is predicted based on visual context and all the previous textual contexts. Unidirectional LSTM cannot generate contextually well formed captions. Moreover, recent object detection and classification methods [14, 13] show that deep, hierarchical methods are better at learning than shallower ones. Wang et al. [2] proposed a deep bidirectional LSTM-based method for image captioning. This method is capable of generating contextually and semantically rich image captions. The proposed architecture consists of a CNN and two separate LSTM networks. It can utilize both past and future context information to learn long term visual language interactions.

Compositional Architecture-Based Image captioning

Compositional architecture-based methods composed of several independent functional building blocks: First, a CNN is used to extract the semantic concepts from the image. Then a language model is used to generate a set of candidate captions. In generating the final caption, these candidate captions are

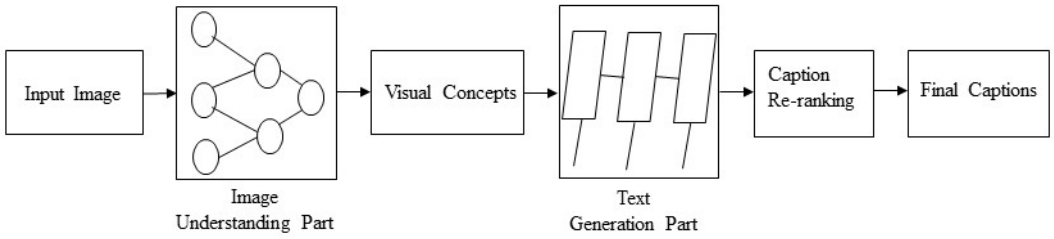


Figure 2.6: A block diagram of a compositional network-based captioning.

re-ranked using a deep multimodal similarity model.
A typical method of this category maintains the following steps:

1. Image features are obtained using a CNN.
2. Visual concepts (e.g. attributes) are obtained from visual features.
3. Multiple captions are generated by a language model using the information of Step 1 and Step 2.
4. The generated captions are re-ranked using a deep multimodal similarity model to select high quality image captions.

A common block diagram of compositional network-based image captioning methods is given in Figure 2.6.

Fang et al.[71] introduced generation-based image captioning. It uses visual detectors, a language model, and a multimodal similarity model to train the model on an image captioning dataset. Image captions can contain nouns, verbs, and adjectives. A vocabulary is formed using 1000 most common words from the training captions. The system works with the image sub-regions rather that the full image. Convolutional neural networks (both AlexNet [14] and VGG16Net) are used for extracting features for the sub-regions of an image. The features of sub-regions are mapped with the words of the vocabulary that likely to be contained in the image captions. Multiple instance learning (MIL) [140] is used to train the model for learning discriminative visual signatures of each word. A maximum entropy (ME) [141] language model is used for generating image captions from these words. Generated captions are ranked by a linear weighting of sentence features. Minimum Error rate training (MERT) [142] is used to learn these weights. Similarity between image and sentence can be easily measured using a common vector representation. Image and sentence fragments are mapped with the common vector representation by a deep multimodal similarity model (DMSM). It achieves a significant improvement in choosing high quality image captions.

Until now a significant number of methods have achieved satisfactory progress in generating image captions. The methods use training and testing samples from the same domain. Therefore, there is no certainty that these methods can perform well in open-domain images. Moreover, they are only good at recognizing generic visual content. There are certain key entities such as celebrities and landmarks that are out of their scope. The generated captions of these methods are evaluated on automatic metrics such as BLEU [143], METEOR [144], and CIDEr [145]. These evaluation metrics have already shown good results on these methods. However, in terms of performance there exists a large gap between

the evaluation of the metrics and human judgement of evaluation [146, 147, 53]. If it is considered real life entity information, the performance could be weaker. However, Tran et al. [80] introduced a different image captioning method. This method is capable of generating image captions even for open domain images. It can detect a diverse set of visual concepts and generate captions for celebrities and landmarks. It uses an external knowledge base Freebase [148] in recognizing a broad range of entities such as celebrities and landmarks. A series of human judgments are applied for evaluating the performances of generated captions. In experiments, it uses three datasets: MS COCO, Adobe-MIT FiveK [149], and images from Instagram. The images of MS COCO dataset were collected from the same domain but the images of other datasets were chosen from an open domain. The method achieves notable performances especially on the challenging Instagram dataset.

Ma et al. [81] proposed another compositional network-based image captioning method. This method uses structural words <object, attribute, activity, scene> to generate semantically meaningful descriptions. It also uses a multi-task method similar to multiple instance learning method [71], and multi-layer optimization method [150] to generate structural words. An LSTM encoder-decoder-based machine translation method [130] is then used to translate the structural words into image captions.

Wang et al. [79] proposed a parallel-fusion RNN-LSTM architecture for image caption generation. The architecture of the method divides the hidden units of RNN and LSTM into a number of same-size parts. The parts work in parallel with corresponding ratios to generate image captions.

2.3.5 Others

Attention-based, Semantic concept-based, Novel object-based methods, and Stylized captions are put together into “Others" group because these categories are independent to other methods.

Attention based Image Captioning

Neural encoder-decoder based approaches were mainly used in machine translation [130]. Following these trends, they have also been used for the task of image captioning and found very effective. In image captioning, a CNN is used as an encoder to extract the visual features from the input image and an RNN is used as a decoder to convert this representation word-by-word into natural language description of the image. However, these methods are unable to analyze the image over time while they generate the descriptions for the image. In addition to this, the methods do not consider the spatial aspects of the image that is relevant to the parts of the image captions. Instead, they generate captions considering the scene as a whole. Attention based mechanisms are becoming increasingly popular in deep learning because they can address these limitations. They can dynamically focus on the various parts of the input image while the output sequences are being produced.

A typical method of this category adopts the following steps:

1. Image information is obtained based on the whole scene by a CNN.
2. The language generation phase generates words or phrases based on the output of Step 1.
3. Salient regions of the given image are focused in each time step of the language generation model based on generated words or phrases.

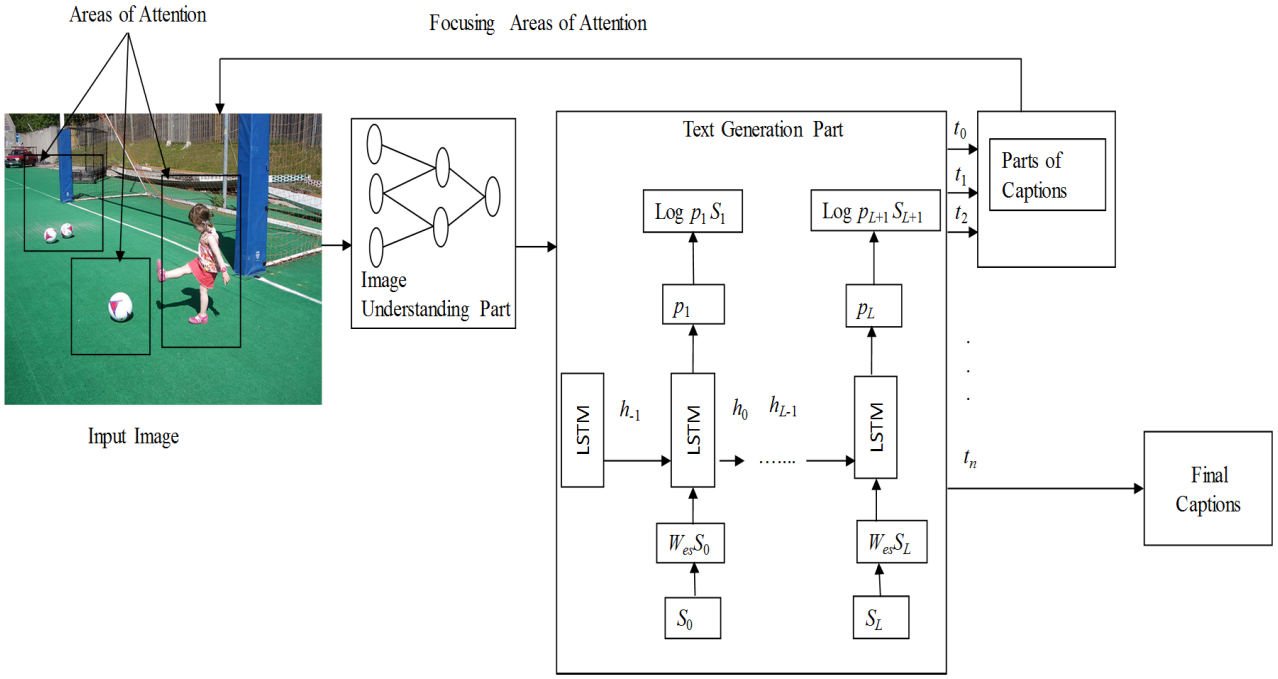


Figure 2.7: A block diagram of a typical attention-based image captioning technique.

4. Captions are updated dynamically until the end state of language generation model.

A block diagram of the attention-based image captioning method is shown in Figure 2.7.

Xu et al. [29] were the first to introduce an attention-based image captioning method. The method describes the salient contents of an image automatically. The main difference between the attention-based methods with other methods is that they can concentrate on the salient parts of the image and generate the corresponding words at the same time. This method applies two different techniques: stochastic hard attention and deterministic soft attention to generate attentions. Most CNN-based approaches use the top layer of ConvNet for extracting information of the salient objects from the image. A drawback of these techniques is that they may lose certain information which is useful to generate detailed captions. In order to preserve the information, the attention method uses features from the lower convolutional layer instead of fully connected layer.

Jin et al. [73] proposed another attention-based image captioning method. This method is capable to extract the flow of abstract meaning based on the semantic relationship between visual information and textual information. It can also obtain higher level semantic information by proposing a scene specific context. The main difference between this method with other attention-based methods is that it introduces multiple visual regions of an image at multiple scales. This technique can extract proper visual information of a particular object. For extracting scene specific context, it first uses the Latent Dirichlet Allocation (LDA) [151] for generating a dictionary from all the captions of the dataset. Then a multilayer perceptron is used to predict a topic vector for every image. A scene factored LSTM that has two stacked layers are used to generate a description for the overall context of the image.

Wu et al. [74] proposed a review-based attention method for image captioning. It introduces a review model that can perform multiple review steps with attention on CNN hidden states. The output of

the CNN is a number of fact vectors that can obtain the global facts of the image. The vectors are given as input to the attention mechanism of the LSTM. For example, a reviewer module can first review: What are the objects in the image? Then it can review the relative positions of the objects and another review can extract the information of the overall context of the image. These information is passed to the decoder to generate image captions.

Pedersoli et al. [87] proposed an area based attention mechanism for image captioning. Previous attention based methods map image regions only to the state of RNN language model. However, this approach associates image regions with caption words given the RNN state. It can predict the next caption word and corresponding image region in each time-step of RNN. It is capable of predicting the next word as well as corresponding image regions in each time-step of RNN for generating image captions. In order to find the areas of attention, previous attention-based image caption methods use either the position of CNN activation grid or object proposals. In contrast, this method uses an end to end trainable convolutional spatial transformer along with CNN activation gird and object proposal methods. A combination of these techniques help this method to compute image adaptive areas of attention. In experiments, the method shows that this new attention mechanism together with the spatial transformer network can produce high quality image captions.

Lu et al. [84] proposed another attention-based image captioning method. The method is based on adaptive attention model with a visual sentinel. Current attention-based image captioning methods focus on the image in every time step of RNN. However, there are some words or phrase (for example: a, of) that do not need to attend visual signals. Moreover, these unnecessary visual signals could affect the caption generation process and degrade the overall performance. Therefore, their proposed method can determine when it will focus on image region and when it will just focus on language generation model. Once it determines to look on the image then it must have to choose the spatial location of the image. The first contribution of this method is to introduce a novel spatial attention method that can compute spatial features from the image. Then in their adaptive attention method, they introduced a new LSTM extension. Generally, an LSTM works as a decoder that can produce a hidden state at every time step. However, this extension is capable of producing an additional visual sentinel that provides a fallback option to the decoder. It also has a sentinel gate that can control how much information the decoder will get from the image.

While attention-based methods look to find the different areas of the image at the time of generating words or phrases for image captions, the attention maps generated by these methods cannot always correspond to the proper region of the image. It can affect the performance of image caption generation. Liu et al. [91] proposed a method for neural image captioning. This method can evaluate and correct the attention map at time step. Correctness means to make consistent map between image regions and generated words. In order to achieve these goals, this method introduced a quantitative evaluation metric to compute the attention maps. It uses Flickr30k entity dataset [152] and MS COCO [153] dataset for measuring both ground truth attention map and semantic labelings of image regions. In order to learn a better attention function, it proposed supervised attention model. Two types of supervised attention models are used here: strong supervision with alignment annotation and weak supervision with semantic labelling. In strong supervision with alignment annotation model, it can directly map ground truth word to a region. However, ground truth alignment is not always possible

because collecting and annotating data is often very expensive. Weak supervision is performed to use bounding box or segmentation masks on MS COCO dataset. In experiments, the method shows that supervised attention model performs better in mapping attention as well as image captioning.

Chen et al. [85] proposed another attention-based image captioning method. This method considers both spatial and channel wise attentions to compute an attention map. The existing attention-based image captioning methods only consider spatial information for generating an attention map. A common drawback of these spatial attention methods are that they compute weighted pooling only on attentive feature map. As a result, these methods lose the spatial information gradually. Moreover, they use the spatial information only from the last conv-layer of the CNN. The receptive field regions of this layer are quite large that make the limited gap between the regions. Therefore, they do not get significant spatial attentions for an image. However, in this method, CNN features are extracted not only from spatial locations but also from different channels and multiple layers. Therefore, it gets significant spatial attention. In addition to this, in this method, each filter of a convolutional layer acts as semantic detectors [154] while other methods use external sources for obtaining semantic information.

In order to reduce the gap between human generated description and machine generated description Tavakoli et al. [42] introduced an attention-based image captioning method. This is a bottom up saliency based attention model that can take advantages for comparisons with other attention-based image captioning methods. It found that humans first describe the more important objects than less important ones. It also shows that the method performs better on unseen data.

Most previous image captioning methods applied top-down approach for constructing a visual attention map. These mechanisms typically focused on some selective regions obtained from the output of one or two layers of a CNN. The input regions are of the same size and have the same shape of receptive field. This approach has a little consideration to the content of the image. However, the method of Anderson et al. [155] applied both top down and bottom up approaches. The bottom up attention mechanism uses Faster R-CNN [111] for region proposals that can select salient regions of an image . Therefore, this method can attend both object level regions as well as other salient image regions.

Park et al. [89] introduced a different type of attention-based image captioning method. This method can generate image captions addressing personal issues of an image. It mainly considers two tasks : hashtag prediction and post generation. This method uses a Context Sequence Memory Network (CSMN) to obtain the context information from the image. Description of an image from personalized view has a lot of applications in social media networks. For example, everyday people share a lot of images as posts in Facebook, Instagram or other social media. Photo-taking or uploading is a very easy task. However, describing them is not easy because it requires theme, sentiment, and context of the image. Therefore, the method considers the past knowledge about the user's vocabularies or writing styles from the prior documents for generating image descriptions. In order to work with this new type of image captioning, the CSMN method has three contributions: first, the memory of this network can work as a repository and retain multiple types of context information. Second, the memory is designed in such a way that it can store all the previously generated words sequentially. As a result, it does not suffer from vanishing gradient problem. Third, the proposed CNN can correlate with multiple memory

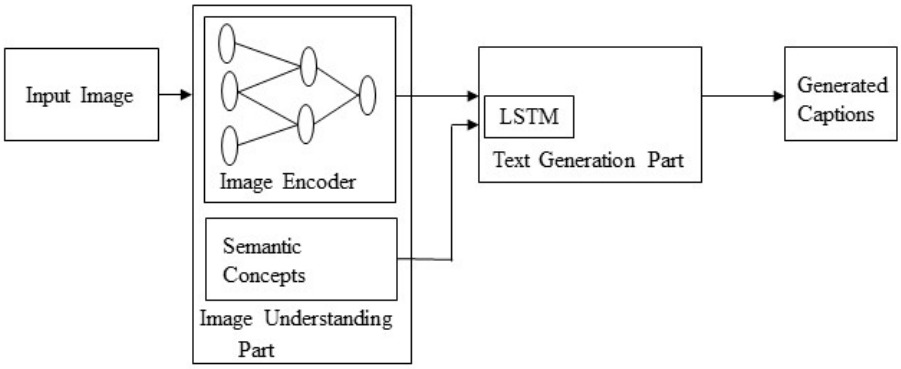


Figure 2.8: A block diagram of a semantic concept-based image captioning.

slots that is helpful for understanding contextual concepts.

Attention-based methods have already shown good performance and efficiency in image captioning as well as other computer vision tasks. However, attention maps generated by these attention based methods are only machine dependent. They do not consider any supervision from human attention. This creates the necessity to think about the gaze information whether it can improve the performance of these attention methods in image captioning. Gaze indicates the cognition and perception of humans about a scene. Human gaze can identify the important locations of objects in an image. Thus, gaze mechanisms have already shown their potential performances in eye-based user modeling [156, 157, 158, 159, 160], object localization [161] or recognition [162] and holistic scene understanding [163, 164]. However, Sugano et al. [75] claimed that gaze information has not yet been integrated in image captioning methods. This method introduced human gaze with the attention mechanism of deep neural networks in generating image captions. The method incorporates human gaze information into an attention-based LSTM model [29]. For experiments, it uses SALICON dataset [165] and achieves good results.

Semantic Concept-Based Image Captioning

Semantic concept-based methods selectively attend to a set of semantic concept proposals extracted from the image. These concepts are then combined into hidden states and the outputs of recurrent neural networks.

The methods in this category follow the following steps:

1. CNN based encoder is used to encode the image features and semantic concepts.
2. Image features are fed into the input of language generation model.
3. Semantic concepts are added to the different hidden states of the language model.
4. The language generation part produces captions with semantic concepts.

A typical block diagram of this category is shown in Figure 2.8.

Karpathy et al. extended their method [68] in [1]. The later method can generate natural language descriptions for both images as well as for their regions. This method employs a novel combination of

CNN over the image regions, bidirectional Recurrent Neural Networks over sentences, and a common multimodal embedding that associates the two modalities. It also demonstrates a multimodal recurrent neural network architecture that utilizes the resultant alignments to train the model for generating novel descriptions of image regions. In this method, dependency tree relations (DTR) are used to train to map the sentence segments with the image regions that have a fixed window context. In contrast to their previous method, this method uses a bidirectional neural network to obtain word representations in the sentence. It considers contiguous fragments of sentences to align in embedding space which is more meaningful, interpretable, and not fixed in length. Generally an RNN considers the current word and the contexts from all the previously generated words for estimating a probability distribution of the next word in a sequence. However, this method extends it for considering the generative process on the content of an input image. This addition is simple but it makes it very effective for generating novel image captions.

Attributes of an image are considered as rich semantic cues. The method of Yao et al. [64] has different architectures to incorporate attributes with image representations. Mainly, two types of architectural representations are introduced here. In the first group, it inserts only attributes to the LSTM or image representations to the LSTM first and then attributes and vice versa. In the second group, it can control the time step of LSTM. It decides whether image representation and attributes will be inputted once or every time step. These variants of architectures are tested on MS COCO dataset and common evaluation metrics.

You et al. [65] proposed a semantic attention-based image captioning method. The method provides a detailed, coherent description of semantically important objects. The top-down paradigms [70, 28, 69, 1, 166, 29, 167] are used for extracting visual features first and then convert them into words. In bottom up approaches, [51, 53, 52, 55, 56, 168] visual concepts (e.g., regions, objects, and attributes) are extracted first from various aspects of an image and then combine them. Fine details of an image are often very important for generating a description of an image. Top- down approaches have limitations in obtaining fine details of the image. Bottom up approaches are capable of operating on any image resolution and therefore they can do work on fine details of the image. However, they have problems in formulating an end to end process. Therefore, semantic based attention model applied both top-down and bottom up approaches for generating image captions. In top-down approaches, the image features are obtained using the last 1024-dimensional convolutional layer of the GoogleNet [15] CNN model. The visual concepts are collected using different non-parametric and parametric method. Nearest neighbour image retrieval technique is used for computing non-parametric visual concepts. Fully convolutional network (FCN) [169] is used to learn attribute from local patches for parametric attribute prediction. Although Xu et al. [29] considered attention-based captioning, it works on fixed and pre-defined spatial location. However, this semantic attention-based method can work on any resolution and any location of the image. Moreover, this method also considers a feedback process that accelerates to generate better image captions.

Previous image captioning methods do not include high level semantic concepts explicitly. However, Wu et al. [101] proposed a high-level semantic concept-based image captioning. It uses an intermediate attribute prediction layer in a neural network-based CNN-LSTM framework. First, attributes are extracted by a CNN-based classifier from training image captions. Then these attributes are used as

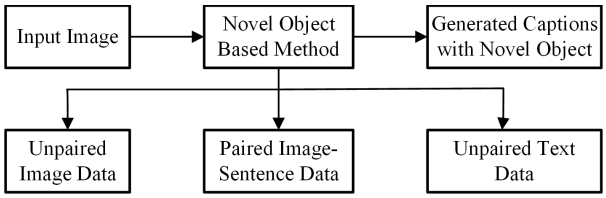


Figure 2.9: A block diagram of a typical novel object-based image captioning.

high level semantic concepts in generating semantically rich image captions.

Recent semantic concept based image captioning methods [101, 65] applied semantic-concept-detection process [112]to obtain explicit semantic concepts. They use these high level semantic concepts in CNN-LSTM based encoder-decoder and achieves significant improvements in image captioning. However, they have problems in generating semantically sound captions. They cannot distribute semantic concepts evenly in the whole sentence. For example, Wu et al. [101] consider the initial state of the LSTM to add semantic concepts. Moreover, it encodes visual features vector or an inferred scene vector from the CNN and then feeds them to LSTM for generating captions. However, Gan et al. [86] introduced a Semantic Compositional Network (SCN) for image captioning. In this method, a semantic concept vector is constructed from all the probable concepts (called tags here) found in the image. This semantic vector has more potential than visual feature vector and scene vector and can generate captions covering the overall meaning of the image. This is called compositional network because it can compose most semantic concepts.

Existing LSTM based image captioning methods have limitations in generating a diverse set of captions because they have to predict the next word on a predefined word by word format. However, a combination of attributes, subjects and their relationship in a sentence irrespective of their location can generate a broad range of image captions. Wang et al. [90] proposed a method that locates the objects and their interactions first and then identifies and extracts the relevant attributes to generate image captions. The main aim of this method is to decompose the ground truth image captions into two parts: Skeleton sentence and attribute phrases. The method is also called Skeleton Key. The architecture of this method has ResNet [16] and two LSTMs called Skel-LSTM and Attr-LSTM. During training, skeleton sentences are trained by Skel-LSTM network and attribute phrases are trained by the Attr-LSTM network. In the testing phase, skeleton sentences are generated first that contain the words for main objects of the image and their relationships. Then these objects look back through the image again to obtain the relevant attributes. It is tested on MS COCO dataset and a new Stock3M dataset and can generate more accurate and novel captions.

Novel Object-based Image Captioning

Despite recent deep learning-based image captioning methods have achieved promising results, they largely depend on the paired image and sentence caption datasets. These type of methods can only generate description of the objects within the context. Therefore, the methods require a large set of training image-sentence pairs. Novel object-based image captioning methods can generate descriptions of novel objects which are not present in paired image-captions datasets.

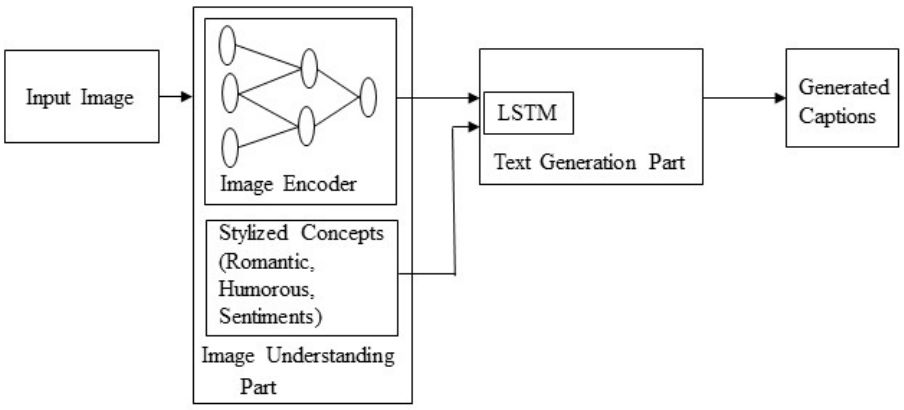


Figure 2.10: A block diagram of image captioning based on different styles.

The methods of this category follow the following general steps:

1. A separate lexical classifier and a language model are trained on unpaired image data and unpaired text data.
2. A deep caption model is trained on paired image caption data.
3. Finally, both models are combined together to train jointly in that can generate captions for novel object.

A simple block diagram of a novel object-based image captioning method is given in Figure 2.9.

Current image captioning methods are trained on image-captions paired datasets. As a result, if they get unseen objects in the test images, they cannot present them in their generated captions. Anne et al. [83] proposed a Deep Compositional Captioner (DCC) that can represent the unseen objects in generated captions.

Yao et al. [97] proposed a copying mechanism to generate description for novel objects. This method uses a separate object recognition dataset to develop classifiers for novel objects. It integrates the appropriate words in the output captions by a decoder RNN with copying mechanism. The architecture of the method adds a new network to recognize the unseen objects from unpaired images and incorporate them with LSTM to generate captions.

Generating captions for the unseen images is a challenging research problem. Venugopalan et al. [99] introduced a Novel Object Captioner (NOC) for generating captions for unseen objects in the image. They used external sources for recognizing unseen objects and learning semantic knowledge.

Stylized Caption

Existing image captioning systems generate captions just based on only the image content that can also be called factual description. They do not consider the stylized part of the text separately from other linguistic patterns. However, the stylized captions can be more expressive and attractive than just only the flat description of an image.

The methods of this category follow the following general steps:

1. CNN based image encoder is used to obtain the image information.
2. A separate text corpus is prepared to extract various stylized concepts (For example: romantic, humorous) from training data.
3. The language generation part can generate stylized and attractive captions using the information of Step 1 and Step 2.

A simple block diagram of stylized image captioning is given in Figure 2.10.

Such captions have become popular because they are particularly valuable for many real-world applications. For example, everyday people are uploading a lot of photos in different social media. The photos need stylized and attractive descriptions. Gan et al. [92] proposed a novel image captioning system called StyleNet. This method can generate attractive captions adding various styles. The architecture of this method consists of a CNN and a factored LSTM that can separate factual and style factors from the captions. It uses multitask sequence to sequence training [170] for identifying the style factors and then add these factors at run time for generating attractive captions. More interestingly, it uses an external monolingual stylized language corpus for training instead of paired images. However, it uses a new stylized image caption dataset called FlickrStyle10k and can generate captions with different styles.

Existing image captioning methods consider the factual description about the objects, scene, and their interactions of an image in generating image captions. In our day to day conversations, communications, interpersonal relationships, and decision making we use various stylized and non-factual expressions such as emotions, pride, and shame. However, Mathews et al. [76] claimed that automatic image descriptions are missing this non-factual aspects. Therefore, they proposed a method called SentiCap. This method can generate image descriptions with positive or negative sentiments. It introduces a novel switching RNN model that combines two CNN+RNNs running in parallel. In each time step, this switching model generates the probability of switching between two RNNs. One generates captions considering the factual words and other considers the words with sentiments. It then takes inputs from the hidden states of both two RNNs for generating captions. This method can generate captions successfully given the appropriate sentiments.

2.3.6 LSTM vs. Others

Image captioning intersects computer vision and natural language processing (NLP) research. NLP tasks, in general, can be formulated as a sequence to sequence learning. Several neural language models such as neural probabilistic language model [171], log-bilinear models [172], skip-gram models [173], and recurrent neural networks (RNNs) [174] have been proposed for learning sequence to sequence tasks. RNNs have widely been used in various sequence learning tasks. However, traditional RNNs suffer from vanishing and exploding gradient problems and cannot adequately handle long-term temporal dependencies.

LSTM [24] networks are a type of RNN that has special units in addition to standard units. LSTM units use a memory cell that can maintain information in memory for long periods of time. In recent years, LSTM based models have dominantly been used in sequence to sequence learning tasks. Another

network, Gated Recurrent Unit (GRU) [30] has a similar structure to LSTM but it does not use separate memory cells and uses fewer gates to control the flow of information.

However, LSTMs ignore the underlying hierarchical structure of a sentence. They also require significant storage due to long-term dependencies through a memory cell. In contrast, CNNs can learn the internal hierarchical structure of the sentences and they are faster in processing than LSTMs. Therefore, recently, convolutional architectures are used in other sequence to sequence tasks, e.g., conditional image generation [32] and machine translation [33, 34, 35].

Inspired by the above success of CNNs in sequence learning tasks, Gu et al. [96] proposed a CNN language model-based image captioning method. This method uses a language-CNN for statistical language modelling. However, the method cannot model the dynamic temporal behaviour of the language model only using a language-CNN. It combines a recurrent network with the language-CNN to model the temporal dependencies properly.

Aneja et al. [46] proposed a convolutional architecture for the task of image captioning. They use a feed-forward network without any recurrent function. The architecture of the method has four components: (i) input embedding layer (ii) image embedding layer (iii) convolutional module, and (iv) output embedding layer. It also uses an attention mechanism to leverage spatial image features. They evaluate their architecture on the challenging MSCOCO dataset and shows comparable performance to an LSTM based method on standard metrics.

Wang et al. [102] proposed another CNN+CNN based image captioning method. It is similar to the method of Aneja et al. except that it uses a hierarchical attention module to connect the vision-CNN with the language-CNN. The authors of this method also investigate the use of various hyperparameters, including the number of layers and the kernel width of the language-CNN. They show that the influence of the hyperparameters can improve the performance of the method in image captioning.

2.4 Datasets and Evaluation Metrics

A number of datasets are used for training, testing, and evaluation of the image captioning methods. The datasets differ in various perspective such as the number of images, the number of captions per image, format of the captions, and image size. Three datasets: Flickr8k [60], Flickr30k [152], and MS COCO Dataset [153] are popularly used. These datasets together with others are described in Section 2.4.1. In this section, we show sample images with their captions generated by image captioning methods on MS COCO, Flickr30k, and Flickr8k datasets. A number of evaluation metrics are used to measure the quality of the generated captions compared to the ground-truth. Each metric applies its own technique for computation and has distinct advantages. The commonly used evaluation metrics are discussed in Section 2.4.2. A summary of deep learning-based image captioning methods with their datasets and evaluation metrics are listed in Table 2.2.

Ground Truth Caption: Two brown bears playing in a field together.

Generated Caption: Two brown bears playing on top of a lush green field.

Ground Truth Caption: A plate of breakfast food with a silver tea pot.

Generated Caption: A close up of a plate of food with a folk and a knife on a table.

Figure 2.11: Captions generated by Wu et al. [175] on some sample images from the MS COCO dataset.

Generated Caption: A young baseball player is sliding into a base.

Generated Caption: A young boy playing with a soccer ball in a field.

Figure 2.12: Captions generated by Chen et al. [176] on some sample images from the Flickr30k dataset.

2.4.1 Datasets

MS COCO Dataset

Microsoft COCO Dataset [153] is a very large dataset for image recognition, segmentation, and captioning. There are various features of MS COCO dataset such as object segmentation, recognition in context, multiple objects per class, more than 300,000 images, more than 2 million instances, 80 object categories, and 5 captions per image. Many image captioning methods [73, 74, 80, 2, 65, 92, 87, 88, 93, 94, 175] use the dataset in their experiments. For example, Wu et al. [175] use MS COCO dataset in their method and the generated captions of two sample images are shown in Figure 2.11.

Flickr30K Dataset

Flickr30K [152] is a dataset for automatic image description and grounded language understanding. It contains 30k images collected from Flickr with 158k captions provided by human annotators. It does not provide any fixed split of images for training, testing, and validation. Researchers can choose their own choice of numbers for training, testing, and validation. The dataset also contains detectors for common objects, a color classifier, and a bias towards selecting larger objects. Image captioning methods such as [1, 28, 2, 101, 176] use this dataset for their experiments. For example, performed their experiment on Flickr30k dataset. The generated captions by Chen et al. [176] of two sample images of the dataset are shown in Figure 2.12.

Reference	Datasets	Evaluation Metrics
Kiros et al. 2014 [66]	IAPR TC-12,SBU	BLEU, PPLX
Kiros et al. 2014 [63]	Flickr 8K, Flickr 30K	R@K, mrank
Mao et al. 2014 [67]	IAPR TC-12, Flickr 8K/30K	BLEU, R@K, mrank
Karpathy et al. 2014 [68]	PASCAL1K, Flickr 8K/30K	R@K, mrank
Mao et al. 2015 [69]	IAPR TC-12, Flickr 8K/30K, MS COCO	BLEU, R@K, mrank
Chen et al. 2015 [70]	PASCAL, Flickr 8K/30K, MS COCO	BLEU, METEOR, CIDEr
Fang et al. 2015 [71]	PASCAL, MS COCO	BLEU, METEOR, PPLX
Jia et al. 2015 [72]	Flickr 8K/30K, MS COCO	BLEU, METEOR, CIDEr
Karpathy et al. 2015 [1]	Flickr 8K/30K, MS COCO	BLEU, METEOR, CIDEr
Vinyals et al. 2015 [28]	Flickr 8K/30K, MS COCO	BLEU, METEOR, CIDEr
Xu et al. 2015 [29]	Flickr 8K/30K, MS COCO	BLEU, METEOR
Jin et al. 2015 [73]	Flickr 8K/30K, MS COCO	BLEU, METEOR, ROUGE, CIDEr
Wu et al. 2016 [74]	MS COCO	BLEU, METEOR, CIDEr
Sugano et at. 2016 [75]	MS COCO	BLEU, METEOR, ROUGE, CIDEr
Mathews et al. 2016 [76]	MS COCO, SentiCap	BLEU, METEOR, ROUGE, CIDEr
Wang et al. 2016 [2]	Flickr 8K/30K, MS COCO	BLEU, R@K
Johnson et al. 2016 [77]	Visual Genome	METEOR, AP, IoU
Mao et al. 2016 [78]	ReferIt	BLEU, METEOR, CIDEr
Wang et al. 2016 [79]	Flickr 8K	BLEU, PPL, METEOR
Tran et al. 2016 [80]	MS COCO, Adobe-MIT, Instagram	Human Evaluation
Ma et al. 2016 [81]	Flickr 8k, UIUC	BLEU, R@K
You et al. 2016 [65]	Flickr 30K, MS COCO	BLEU, METEOR, ROUGE, CIDEr
Yang et al. 2016 [82]	Visual Genome	METEOR, AP, IoU
Anne et al. 2016 [83]	MS COCO, ImageNet	BLEU, METEOR
Yao et al. 2017 [64]	MS COCO	BLEU, METEOR, ROUGE, CIDEr
Lu et al. 2017 [84]	Flickr 30K, MS COCO	BLEU, METEOR, CIDEr
Chen et al. 2017 [85]	Flickr 8K/30K, MS COCO	BLEU, METEOR, ROUGE, CIDEr
Gan et al. 2017 [86]	Flickr 30K, MS COCO	BLEU, METEOR, CIDEr
Pedersoli et al. 2017 [87]	MS COCO	BLEU, METEOR, CIDEr
Ren et al. 2017 [88]	MS COCO	BLEU, METEOR, ROUGE, CIDEr
Park et al. 2017 [89]	Instagram	BLEU, METEOR, ROUGE, CIDEr
Wang et al. 2017 [90]	MS COCO, Stock3M	SPICE, METEOR, ROUGE, CIDEr
Tavakoli et al. 2017 [42]	MS COCO, PASCAL 50S	BLEU, METEOR, ROUGE, CIDEr
Liu et al. 2017 [91]	Flickr 30K, MS COCO	BLEU, METEOR
Gan et al. 2017 [92]	FlickrStyle10K	BLEU, METEOR, ROUGE, CIDEr
Dai et al. 2017 [93]	Flickr 30K, MS COCO	E-NGAN, E-GAN, SPICE, CIDEr
Shetty et al. 2017 [94]	MS COCO	Human Evaluation, SPICE, METEOR
Liu et al. 2017 [95]	MS COCO	SPIDEr, Human Evaluation
Gu et al. 2017 [96]	Flickr 30K, MS COCO	BLEU, METEOR, CIDEr, SPICE
Yao et al. 2017 [97]	MS COCO, ImageNet	METEOR
Rennie et al. 2017 [98]	MS COCO	BLEU, METEOR, CIDEr, ROUGE
Vsub et al. 2017 [99]	MS COCO, ImageNet	METEOR
Zhang et al. 2017 [100]	MS COCO	BLEU, METEOR, ROUGE, CIDEr
Wu et al. 2018 [101]	Flickr 8K/30K, MS COCO	BLEU, METEOR, CIDEr
Aneja et al. 2018 [46]	MS COCO	BLEU, METEOR, ROUGE, CIDEr
Wang et al. 2018 [102]	MS COCO	BLEU, METEOR, ROUGE, CIDEr

Table 2.2: An overview of methods, datasets, and evaluation metrics

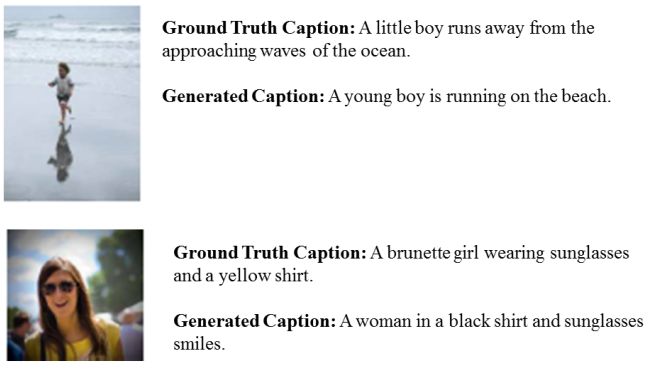


Figure 2.13: Captions generated by Jia et al. [72] on some sample images from the Flickr8k dataset.

Flickr8K Dataset

Flickr8k [60] is a popular dataset and has 8000 images collected from Flickr. The training data consists of 6000 images, the test and development data, each consists of 1,000 images. Each image in the dataset has 5 reference captions annotated by humans. A number of image captioning methods [72, 73, 29, 2, 101, 85] have performed experiments using the dataset. Two sample results by Jia et al. [72] on this dataset are shown in Figure 2.13.

Visual Genome Dataset

Visual Genome dataset [129] is another dataset for image captioning. Image captioning requires not only to recognise the objects of an image but it also needs reasoning their interactions and attributes. Unlike the first three datasets where a caption is given to the whole scene, Visual Genome dataset has separate captions for multiple regions in an image. The dataset has seven main parts: region descriptions, objects, attributes, relationships, region graphs, scene graphs, and question answer pairs. The dataset has more than 108k images. Each image contains an average of 35 objects, 26 attributes, and 21 pairwise relationships between objects.

Instagram Dataset

Tran et al. [80] and Park et al. [89] created two datasets using images from Instagram which is a photo-sharing social networking services. The dataset of Tran et al. has about 10k images which are mostly from celebrities. However, Park et al. used their dataset for hashtag prediction and post-generation tasks in social media networks. This dataset contains 1.1m posts on a wide range of topics and a long hashtag lists from 6.3k users.

IAPR TC-12 Dataset

IAPR TC-12 dataset [177] has 20k images. The images are collected from various sources such as sports, photographs of people, animals, landscapes and many other locations around the world. The images of this dataset have captions in multiple languages. Images have multiple objects as well.

Stock3M Dataset

Stock3M dataset has 3,217,654 images uploaded by users and it is 26 times larger than MSCOCO dataset. The images of this dataset have a diversity of content.

MIT-Adobe FiveK dataset

MIT-Adobe FiveK [149] dataset consists of 5,000 images. These images contain a diverse set of scenes, subjects, and lighting conditions and they are mainly about people, nature, and man-made objects.

FlickrStyle10k Dataset

FlickrStyle10k dataset has 10,000 Flickr images with stylized captions. The training data consists of 7000 images. The validation and test data consists of 2,000 and 1,000 images respectively. Each image contains romantic, humorous, and factual captions.

2.4.2 Evaluation Metrics

BLEU

BLEU (Bilingual evaluation understudy) [143] is a metric that is used to measure the quality of machine generated text. Individual text segments are compared with a set of reference texts and scores are computed for each of them. In estimating the overall quality of the generated text, the computed scores are averaged. However, syntactical correctness is not considered here. The performance of the BLEU metric is varied depending on the number of reference translations and the size of the generated text. Subsequently, Papineni et al. introduced a modified precision metric. This metrics uses n-grams. BLEU is popular because it is a pioneer in automatic evaluation of machine translated text and has a reasonable correlation with human judgements of quality [178, 147]. However, it has a few limitations such as BLEU scores are good only if the generated text is short [147]. There are some cases where an increase in BLEU score does not mean that the quality of the generated text is good [179].

ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [180] is a set of metrics that are used for measuring the quality of text summary. It compares word sequences, word pairs, and n-grams with a set of reference summaries created by humans. Different types of ROUGE such as ROUGE-1, 2, ROUGE-W, ROUGE-SU4 are used for different tasks. For example, ROUGE-1 and ROUGE-W are appropriate for single document evaluation whereas ROUGE-2 and ROUGE-SU4 have good performance in short summaries. However, ROUGE has problems in evaluating multi-document text summary.

METEOR

METEOR (Metric for Evaluation of Translation with Explicit ORdering) [181] is another metric used to evaluate the machine translated language. Standard word segments are compared with the reference texts. In addition to this, stems of a sentence and synonyms of words are also considered for matching. METEOR can make better correlation at the sentence or the segment level.

CIDEr

CIDEr (Consensus-based Image Descripton Evaluation) [145] is an automatic consensus metric for evaluating image descriptions. Most existing datasets have only five captions per image. Previous evaluation metrics work with these small number of sentences and are not enough to measure the consensus between generated captions and human judgement. However, CIDEr achieves human consensus using term frequency-inverse document frequency (TF-IDF) [182].

SPICE

SPICE (Semantic Propositional Image Caption Evaluation) [183] is a new caption evaluation metric based on semantic concept. It is based on a graph-based semantic representation called scene-graph [184, 185]. This graph can extract the information of different objects, attributes and their relationships from the image descriptions.

Existing image captioning methods compute log-likelihood scores to evaluate their generated captions. They use BLEU, METEOR, ROUGE, SPICE, and CIDEr as evaluation metrics. However, BLEU, METEOR, ROUGE are not well correlated with human assessments of quality. SPICE and CIDEr have better correlation but they are hard to optimize. Liu et al. [95] introduced a new captions evaluation metric that is a good choice by human raters. It is developed by a combination of SPICE and CIDEr, and termed as SPIDEr. It uses a policy gradient method to optimize the metrics.

The quality of image captioning depends on the assessment of two main aspects: adequacy and fluency. An evaluation metric needs to focus on a diverse set of linguistic features to achieve these aspects. However, commonly used evaluation metrics consider only some specific features (e.g., lexical or semantic) of languages. Sharif et al. [186] proposed learning-based composite metrics for evaluation of image captions. The composite metric incorporates a set of linguistic features to achieve the two main aspects of assessment and shows improved performances.

2.5 Comparison on benchmark datasets and common evaluation metrics

While formal experimental evaluation was left out of the scope of this paper, we present a brief analysis of the experimental results and the performance of various techniques as reported. We cover three sets of results:

1. We find a number of methods use the first three datasets listed in Section 2.4.1. and a number of commonly used evaluation metrics to present the results. These results are shown in Table 2.3.

2. A few methods fall into the following groups: Attention-based and Other deep learning-based (Reinforcement learning and GAN-based methods) image captioning. The results of such methods are shown in Tables 2.4 and 2.5, respectively.

3. We also list the methods that proivide top two results scored on each evaluation metric on the MSCOCO dataset. These results are shown in Table 2.6.

Dataset	Method	Category	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8k	Mao et al. 2015 [69]	MS,SL,WS	0.565	0.386	0.256	0.170	-
	Jia et al. 2015 [72]	VS,SL,WS,EDA	0.647	0.459	0.318	0.216	0.201
	Xu et al. 2015 [29]	VS,SL,WS,EDA,AB	0.670	0.457	0.314	0.213	0.203
	Wu et al. 2018 [101]	VS,SL,WS,EDA,SCB	0.740	0.540	0.380	0.270	-
Flickr30k	Mao et al. 2015 [69]	MS,SL,WS	0.600	0.410	0.280	0.190	-
	Jia et al. 2015 [72]	VS,SL,WS,EDA	0.646	0.466	0.305	0.206	0.179
	Xu et al. 2015 [29]	VS,SL,WS,EDA,AB	0.669	0.439	0.296	0.199	0.184
	Wu et al. 2018 [101]	VS,SL,WS,EDA,SCB	0.730	0.550	0.400	0.280	-
MSCOCO	Mao et al. 2015 [69]	MS,SL,WS	0.670	0.490	0.350	0.250	-
	Jia et al. 2015 [72]	VS,SL,WS,EDA	0.670	0.491	0.358	0.264	0.227
	Xu et al. 2015 [29]	VS,SL,WS,EDA,AB	0.718	0.504	0.357	0.250	0.230
	Wu et al. 2018 [101]	VS,SL,WS,EDA,SCB	0.740	0.560	0.420	0.310	0.260

Table 2.3: Performance of different image captioning methods on three benchmark datasets and commonly used evaluation metrics.

As shown in Table 2.3, on Flickr8k, Mao et al. achieved 0.565, 0.386, 0.256, and 0.170 on BLEU-1, BLEU-2, BLEU-3, and BLEU-4 , respectively. For Flickr30k dataset, the scores are 0.600, 0.410, 0.280, and 0.190, respectively which are higher than the Flickr8k scores. The highest scores were achieved on the MSCOCO dataset. The higher results on a larger dataset follows the fact that a large dataset has more data, comprehensive representation of various scenes, complexities, and their own natural context.

The results of Jia et al. are similar for Flickr8k and Flickr30k datasets but higher on MSCOCO dataset. The method uses visual space for mapping image-features and text features. Mao et al. use multimodal space for the mapping of image-features and text features. On the other hand, Jia et al. use visual space for the mapping. Moreover, the method uses an Encoder-Decoder architecture where it can guide the decoder part dynamically. Consequently, this method performs better than Mao et al.

Xu et al. also perform better on MSCOCO dataset. This method outperformed both Mao et al. and Jia et al. The main reason behind this is that it uses an attention mechanism which focuses only on relevant objects of the image. The semantic concept-based methods can generate semantically rich captions. Wu et al. proposed a semantic concept-based image captioning method. This method first predicts the attributes of different objects from the image and then adds these attributes with the captions which are semantically meaningful. In terms of performance, the method is superior to all the methods mentioned in Table 2.3.

Table 2.4 shows the results of attention-based based methods on MSCOCO dataset. Xu et al.'s stochastic hard attention produced better results than deterministic soft attention. However, these results were outperformed by Jin et al. which can update its attention based on the scene-specific context.

Wu et al. 2016 and Pedersoli et al. 2017 only show BLEU-4 and METEOR scores which are higher than the aforementioned methods. The method of Wu et al. uses an attention mechanism with a review process. The review process checks the focused attention in every time step and updates it if necessary. This mechanism helps to achieve better results than the prior attention-based methods. Pedersoli et al. propose a different attention mechanism that maps the focused image regions directly with the caption words instead of LSTM state. This behavior of the method drives it to achieve top

Method	Category	MS COCO						
		BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
Xu et al. 2015 [29], soft	VS,SL,WS,EDA,VC	0.707	0.492	0.344	0.243	0.239	-	-
Xu et al. 2015 [29], hard	VS,SL,WS,EDA,VC	0.718	0.504	0.357	0.250	0.230	-	-
Jin et al. 2015 [73]	VS,SL,WS,EDA,VC	0.697	0.519	0.381	0.282	0.235	0.509	0.838
Wu et al. 2016 [74]	VS,SL,WS,EDA,VC	-	-	-	0.290	0.237	-	0.886
Pedersoli et al. 2017 [87]	VS,SL,WS,EDA,VC	-	-	-	0.307	0.245	-	0.938

Table 2.4: Performance of attention-based image captioning methods on MSCOCO dataset and commonly used evaluation metrics.

Method	Category	MS COCO							
		BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr	SPICE
Shetty et al. 2017_{GAN} [94]	VS,ODL,WS,EDA	-	-	-	-	0.239	-	-	0.167
Ren et al. 2017_{RL} [88]	VS,ODL,WS,EDA	0.713	0.539	0.403	0.304	0.251	0.525	0.937	-
Zhang et al. 2017_{RL} [100]	VS,ODL,WS,EDA	-	-	-	0.344	0.267	0.558	1.162	-

Table 2.5: Performance of Other Deep Learning-based image captioning methods on MSCOCO dataset and commonly used evaluation metrics.

performances among the mentioned attention-based methods in Table 2.4.

Reinforcement learning-based (RL) and GAN-based methods are becoming increasingly popular. We name them as “Other Deep Learning-based Image Captioning". The results of the methods of this group are shown in Table 2.5. The methods do not have results on commonly used evaluation metrics. However, they have their own potentials to generate the descriptions for the image.

Shetty et al. employed adversarial training in their image captioning method. This method is capable to generate diverse captions. The captions are less-biased with the ground-truth captions compared to the methods use maximum likelihood estimation. To take the advantages of RL, Ren et al. proposed a method that can predict all possible next words for the current word in current time step. This mechanism helps them to generate contextually more accurate captions. Actor-critic of RL are similar to the Generator and the Discriminator of GAN. However, at the beginning of the training, both actor and critic do not have any knowledge about data. Zhang et al. proposed an actor-critic-based image captioning method. This method is capable of predicting the ultimate captions at its early stage and can generate more accurate captions than other reinforcement learning-based methods.

We found that the performance of a technique can vary across different metrics. Table 2.6 shows the methods based on the top two scores on every individual evaluation metric. For example, Lu et al., Gan et al., and Zhang et al. are within the top two methods based on the scores achieved on BLEU-n and METEOR metrics. BLEU-n metrics use variable length phrases of generated captions to match against ground-truth captions. METEOR [181] considers the precision, recall, and the alignments of the matched tokens. Therefore, the generated captions by these methods have good precision and recall accuracy as well as the good similarity in word level. ROUGE-L evaluates the adequacy and fluency of generated captions, whereas CIDEr focuses on grammaticality and saliency. SPICE can analyse the semantics of the generated captions. Zhang et al., Rennie et al., and Lu et al. can generate captions, which have adequacy, fluency, saliency, and are grammaticality correct than other methods in Table 2.6. Gu et al. and Yao et al. perform well in generating semantically correct captions.

**A dash (-) in the tables of this paper indicates results are unavailable*

Method	Category	MSCOCO							
		BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr	SPICE
Lu et al. 2017 [84]	VS,SL,WS,EDA,AB	***0.742***	***0.580***	**0.439**	0.332	**0.266**	-	**1.085**	-
Gan et al. 2017 [86]	VS,SL,WS,CA,SCB	**0.741**	**0.578**	***0.444***	**0.341**	0.261	-	1.041	-
Zhang et al. 2017 [100]	VS,ODL,WS,EDA	-	-	-	***0.344***	***0.267***	***0.558***	***1.162***	-
Rennie et al. 2017 [98]	VS,ODL,WS,EDA	-	-	-	.319	0.255	**0.543**	1.06	-
Yao et al. 2017 [64]	VS,SL,WS,EDA,SCB	0.734	0.567	0.430	0.326	0.254	0.540	1.00	***0.186***
Gu et al. 2017 [96]	VS,SL,WS,EDA	0.720	0.550	0.410	0.300	0.240	-	0.960	**0.176**

Table 2.6: Top two methods based on different evaluation metrics and MSCOCO dataset (Bold and Italic indicates the best result; Bold indicates the second best result).

2.6 Discussions and Future Research Directions

Many deep learning-based methods have been proposed for generating automatic image captions in the recent years. Supervised learning, reinforcement learning, and GAN based methods are commonly used in generating image captions. Both visual space and multimodal space can be used in supervised learning-based methods. The main difference between visual space and multimodal space occurs in mapping. Visual space-based methods perform explicit mapping from images to descriptions. In contrast, multimodal space-based methods incorporate implicit vision and language models. Supervised learning-based methods are further categorized into Encoder-Decoder architecture-based, Compositional architecture-based, Attention-based, Semantic concept-based, Stylized captions, Dense image captioning, and Novel object-based image captioning.

Encoder-Decoder architecture-based methods use a simple CNN and a text generator for generating image captions. Attention-based image captioning methods focus on different salient parts of the image and achieve better performance than encoder-decoder architecture-based methods. Semantic concept-based image captioning methods selectively focus on different parts of the image and can generate semantically rich captions. Dense image captioning methods can generate region based image captions. Stylized image captions express various emotions such as romance, pride, and shame. GAN and RL based image captioning methods can generate diverse and multiple captions.

MSCOCO, Flickr30k and Flickr8k dataset are common and popular datasets used for image captioning. MSCOCO dataset is very large dataset and all the images in these datasets have multiple captions. Visual Genome dataset is mainly used for region based image captioning. Different evaluation metrics are used for measuring the performances of image captions. BLEU metric is good for small sentence evaluation. ROUGE has different types and they can be used for evaluating different types of texts. METEOR can perform an evaluation on various segments of a caption. SPICE is better in understanding semantic details of captions compared to other evaluation metrics.

Although success has been achieved in recent years, there is still a large scope for improvement. Generation based methods can generate novel captions for every image. However, these methods fail to detect prominent objects and attributes and their relationships to some extent in generating accurate and multiple captions. In addition to this, the accuracy of the generated captions largely depends on syntactically correct and diverse captions which in turn rely on powerful and sophisticated language generation model. Existing methods show their performances on the datasets where images are collected from the same domain. Therefore, working on open domain dataset will be an interesting avenue for research in this area. Image-based factual descriptions are not enough to generate high-quality captions.

External knowledge can be added in order to generate attractive image captions. Supervised learning needs a large amount of labelled data for training. Therefore, unsupervised learning and reinforcement learning will be more popular in future in image captioning.

2.7 Conclusions

In this paper, we have reviewed deep learning-based image captioning methods. We have given a taxonomy of image captioning techniques, shown generic block diagram of the major groups and highlighted their pros and cons. We discussed different evaluation metrics and datasets with their strengths and weaknesses. A brief summary of experimental results is also given. We briefly outlined potential research directions in this area. Although deep learning-based image captioning methods have achieved a remarkable progress in recent years, a robust image captioning method that is able to generate high quality captions for nearly all images is yet to be achieved. With the advent of novel deep learning network architectures, automatic image captioning will remain an active research area for some time.

Chapter 3

Attention-based Image Captioning Using DenseNet Features

ABSTRACT

We present an attention-based image captioning method using DenseNet features. Conventional image captioning methods depend on visual information of the whole scene to generate image captions. Such a mechanism often fails to get the information of salient objects and cannot generate semantically correct captions. We consider an attention mechanism that can focus on relevant parts of the image to generate fine-grained description of image. We use image features from DenseNet. We conduct our experiments on the MSCOCO dataset and analyse the results using BLEU, METEOR, ROUGE, and CIDEr evaluation metrics. Our proposed method achieved 53.6, 39.8, and 29.5 on BLEU-2, 3, and 4 metrics, respectively, which are superior to the state-of-the-art methods.

3.1 Introduction

Image captioning is the task of describing an image with natural language. Automatic image captioning has many applications such as helping visually impaired people to understand their surroundings and automatic image indexing.

Image captioning has been extensively studied in the literature. It has been addressed using both traditional techniques and deep learning techniques [187, 29]. Deep learning-based techniques such as Convolutional Neural Network (CNN) [10], Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM) [24] have been widely used as they are capable of handling the complexities and challenges of image captioning.

Different CNNs such as AlexNet [14], VGGNet [13], ResNet [16], and DenseNet [17] have their own strengths and weaknesses. It is generally accepted that the deeper is the network, the more relevant are the learned features [16]. However, if the depth of the network exceeds a certain number, one

This chapter is published in the proceedings of the International Conference on Neural Information Processing (ICONIP). Sydney, Australia, 2019, under the title of “Attention-based Image Captioning Using DenseNet Features".

may obtain the opposite effect, i.e., a decline in performance. There are two main reasons behind this fact: (i) The vanishing-gradient problem: when the input or the gradient passes through many layers, it can vanish or "wash out" by the time it reaches the end of the network, and (ii) the degradation problem. This problem has been addressed in the literature by using residual learning mechanisms such as ResNet [17]. However, the element-wise addition is used in identity mapping in ResNet is computationally expensive during training. On the other hand, in DenseNet, each layer has connections with every other layer in the network in a feed-forward manner. The network reuses the feature-maps and uses concatenation for various operations instead of addition. Therefore, it can reduce the number of parameters and it can be memory efficient. Moreover, since each layer of DenseNet receives feature maps from all previous layers, it gets diversified features and tends to have rich patterns. For this reason, we use DenseNet for extracting features from images.

Most existing image captioning methods including deep learning-based techniques focus only on factual description of an image [28, 29]. During feature learning, these methods compress the entire scene into a fixed vector representations. As a result, they often lose information of the prominent objects in the scene. Attention mechanisms [45] can focus on the parts of the image that are relevant, for a period of time, similar to the human visual system. Simultaneously, they can discard irrelevant information.

In this paper, we propose an image captioning method where attention is a key mechanism to describe the important objects in a scene. Overall, the contributions of the paper are:

- We use DenseNet [17] for extracting image features as it can extract diversified and rich feature patterns.
- We use an attention mechanism in our image captioning method that can focus on the salient parts of the image and describe fine-grained captions.

We organize the rest of the paper as follows: In Section 3.2, we discuss the related work. The architecture and methodology are described in Section 3.3. Experiments and results are discussed in Section 3.4. Section 3.5 concludes the paper.

3.2 Related Work

This section is divided into two major parts; (i) image captioning and (ii) attention in image captioning.

3.2.1 Image Captioning

In the last few years, with the advancements in deep neural network models for Computer Vision (CV) and Natural Language Processing (NLP), automatic image captioning has become a promising research area. Hossain et al. [188] present a comprehensive survey of the topic. They grouped the methods into a number of categories. They include template-based image captioning, retrieval-based image captioning, and novel caption generation. Template-based approaches [51] have fixed templates with a number of blank slots to generate captions.

Captions can also be retrieved from visual space and multimodal space. In retrieval-based approaches, captions are retrieved from a set of existing captions. These methods produce general and syntactically correct captions. However, they cannot produce syntactically correct image-specific captions [5].

Novel captions can be generated from both visual space and multimodal space. In most cases, they use deep machine-learning-based techniques. A general approach of this category is to analyze the visual content of the image first and then generate image captions from the visual content using a language model. These methods can generate image captions that are semantically more accurate than the aforementioned approaches [5]. However, these methods have problems in identifying prominent objects from image.

3.2.2 Attention in Image Captioning

Most existing image captioning methods consider the scene as a whole at the time of generating captions. These methods cannot analyze the image over time while they generate the descriptions. Attention-based image captioning methods can dynamically focus on the relevant parts of the image while the output sequences are being produced. Such methods are now getting increasingly popular in deep learning as well as in image captioning.

The first attention-based image captioning method was proposed by Xu et al. [29]. This method can automatically describe the salient contents of an image. It introduces two attention-based generators: stochastic hard attention and deterministic soft attention to describe the main parts of the image.

There are some words or phrases such as "a" and "of" that do not need to attend visual signals. These unnecessary visual signals degradate the overall performance of image captioning. Lu et al. [84] proposed an adaptive attention-based image captioning method. This method uses an LSTM decoder that has a visual sentinel gate. This gate can control how much information the LSTM decoder will get from the image. A different type of attention-based image captioning method was introduced by Park et al. [89], which addresses the personal issues of an image. In fact, nowadays, people share a lot of photos on social media. It mainly focuses on two tasks: hashtag prediction and post generation. It uses a Contex Sequence Memory Network (CSMN) to obtain different types of theme, sentiment, and context from the image. However, these methods have problems in recognizing correct objects from image.

3.3 The Proposed Architecture

The input of our model is an image and the output is description of that image. The proposed model consists of three main parts: a CNN encoder (i.e., DenseNet) to extract image features, an attention module (i.e., Soft and Hard attention) to dynamically focus on the relevant parts of the image and a decoder LSTM to generate image captions with the information of salient objects. The overall architecture of our image captioning method is shown in Figure 3.1. In Figure 3.1, $\hat{z}_t$, h_t and s_t refer to the context vector, the LSTM hidden state vector, and the previously generated word, respectively. The LSTM is trained to compute the output word (s_t) probability condition on the context vector ($\hat{z}_t$) and the previously generated word s_t at time t. It is given as:

$$P(s_0, s_1, \ldots, s_m) = \prod_{i=0}^{m} P(s_i|\hat{z}, s_0, s_1 \ldots, s_m) \tag{3.1}$$

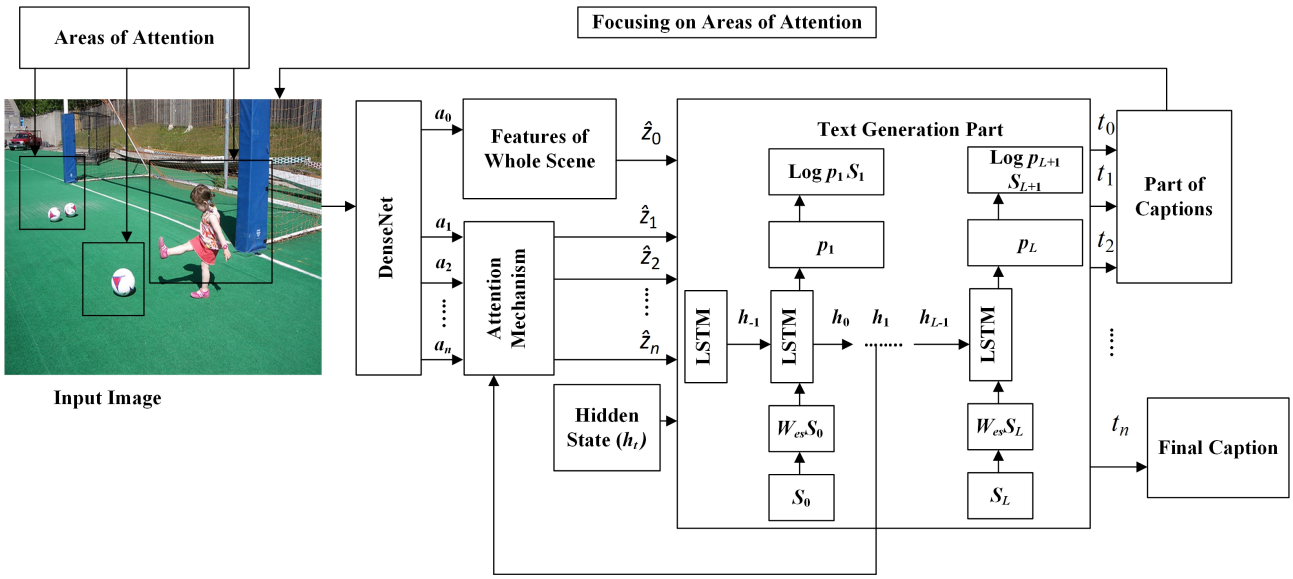


Figure 3.1: The overall architecture of the proposed image captioning method. The method uses DenseNet to obtain the image features and an attention mechanism to selectively focus on relevant parts of the image.

3.3.1 Image Encoder

Traditional convolutional networks with L layers have L connections. However, DenseNet has $L(L+1)/2$ direct connections. As a result, the feature-maps of all preceding layers are used as inputs of the current layer, and its own feature-maps are used as inputs into all subsequent layers. Suppose an image I_0 is passed through a CNN. Consider a CNN network that has L layers and each layer l applies a non-linear transformation $H_l(.)$. In traditional CNNs such as AlexNet, GoogleNet, and VGGNet, the output of the l-th layer is sent to the input of $(l+1)$-th layer. The transformation function for the networks can be written as:

$$I_l = H_l(I_{l-1}) \tag{3.2}$$

However, the transformation mechanism is different in ResNets. Such a network adds a skip-connection using an identity function for transformation:

$$I_l = H_l(I_{l-1}) + I_{l-1} \tag{3.3}$$

In ResNets, the identify function is used to flow gradient from later layers to the earlier layers. Since summation is used here to merge the output of H_l and the identity function, it may delay the information flow to the network. Consequently, the l-th layer of DenseNet receives the feature-maps of all the previous layers, $I_0, I_1, \ldots, I_l$ as input. The transformation function for DenseNet is:

$$I_l = H_l([I_0, I_1, \ldots, I_{l-1}]) \tag{3.4}$$

where $[I_0, I_1, \ldots, I_{l-1}]$ refers to the concatenation of the feature-maps generated in layers $0, 1, \ldots, l-1$ and $H_l(.)$ is a composite function.

3.3.2 Attention Models

Attention is a mechanism that has the ability to weight different regions of the image differently. The attention-based network can add more weights to the salient regions of the image. Moreover, the network can recompute its attention for the relevant parts of the image according to the perceived importance from LSTM. This recomputed image feature is a dynamic representation of the relevant parts of the image and is called context vector ($\hat{z}_t$). Such a vector is computed from the annotation vector a_i defined in equation 3.5 and the attention weight (α_{ti}). The attention weight is obtained from the alignment score (e_{ti}). The score defines how well each annotation vector matches with the previous hidden state output (h_{t-1}) of the LSTM decoder. Such an alignment score is computed by applying an attention function (f_{att}):

$$e_{ti} = f_{\text{att}}(a_i, h_{t-1}) \tag{3.5}$$

Next, the attention weight is obtained by normalizing e_{ti} using a Softmax function:

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})} \tag{3.6}$$

Then we compute the context vector ($\hat{z}_t$) using equations 3.5 and 3.6 as follows:

$$\hat{z}_t = \phi(\{a_i\}, \{\alpha_i\}) \tag{3.7}$$

In equation 3.7, the ϕ is a transformation function that returns a single vector. The attention weights are positive and their summation is equal to one that can also be termed as: $\alpha_t > 0$ and $||\alpha||_t = 1$.

Deterministic Soft Attention: We compute (α_i) for each image region (x_i) and then we calculate the weighted average for (x_i) to use it as the input of LSTM. Hence the context vector $\hat{z}_t$ for soft attention can be calculated as:

$$E_{p(x_t|a)}[\hat{z}_t] = \sum_{i=1}^{L} \alpha_{t,i} a_i \tag{3.8}$$

Stochastic Hard Attention: In hard attention, instead of a weighted average, we use α_i in a stochastic manner to pick up one x_i. We compute $\hat{z}_t$ for hard attention as follows:

$$p(x_{t,i=1}|x_{j<t}, a) = \alpha_{t,i} \tag{3.9}$$

$$\hat{z}_t = \sum_{i=1} s_{t,i}, a_i \tag{3.10}$$

3.3.3 Language Decoder

LSTM is a type of RNN that works well on temporal and sequential data. RNNs are similar to feed forward artificial neural network except that they can feed outputs back to the input. In our model, LSTM takes context vector ($\hat{z}_t$) and the hidden state vector (h_t) as input at each time step and generates a word as output.

3.4 Experiments

In this section, we demonstrate our proposed method using the MSCOCO [153] dataset and commonly used evaluation metrics such as BLEU [143], METEOR [181], ROUGE [180], and CIDEr [145] for image captioning. We implement both stochastic hard attention and deterministic soft attention.

3.4.1 Dataset and Experimental Setup

MSCOCO: Microsoft COCO Dataset [153] is a large and popular dataset for object recognition, segmentation, and image captioning. The dataset consists of 82,783 training and 40,504 validation images. Each image has at least 5 human annotated ground-truth captions. We choose MSCOCO dataset because it has much more images and annotations for both training and testing compared to Flickr8K [60] and Flickr30K [152] datasets.

Implementation Details: In our framework, we use DenseNet121 [17] with fully connected layers for obtaining image features. The DenseNet121 is pre-trained on ImageNet dataset [189]. We apply the fc7 feature map to compute attention features. The dimension of our feature map is 1×1024. The size of the hidden layer in the prediction module is 1024. We apply dropout, learning rate to 0.001 and use a linear layer to obtain a 512-dimensional word embedding. We also apply Adam optimizer [190] with mini-batch size 16 to train the model. Then we upsample the word embedding vector via ReLU activation on the fully connected layer, and pass it through a softmax to obtain the output word probabilities $P_{i,w}(y_i|y_{<i}, I)$. Our method was trained for 20 epochs and we evaluate the metrics on the validation dataset, after every epoch, to pick the best model. The model was implemented in Tensorflow 1.2.

Compared Models: We compare our method with several state-of-the-art image captioning methods: DeepVS [1], m-RNN [69], Google NIC [28], LRCN [166], hard-ATT [29], soft-ATT [29], and ConvCap [46] on MSCOCO dataset and commonly used evaluation metrics in Table 3.1.

Evaluation Metrics: A number of evaluation metrics such as BLEU [143], METEOR [181], ROUGE [180], and CIDEr [145] have widely been used to measure the quality of the generated image captions compared to the ground truth. Each metric applies its own technique for computation and has distinct advantages. In this experiment, we consider BLEU-1, BLEU-2, BLEU-3, BLEU-4, METEOR, ROUGE, and CIDEr to evaluate our method. For all metrics, higher values indicate better performance.

3.4.2 Analysis of Results on MSCOCO dataset

Quantitative Evaluation Results: Table 3.1 shows the results on the MSCOCO dataset. Note that all methods only use the image information without semantics or attributes boosting. The methods use different CNN encoder for image representation. NIC [28] and g-LSTM [72] exploit GoogleNet to extract image features. LRCN [166] utilizes AlexNet to obtain the features. DeepVS [1], m-RNN [69], and Soft/Hard attention [29] use VGGNet to get image-level representation. However, we use DenseNet in our method for the task. In terms of the BLEU-1 score, which only considers bigrams, the methods NIC [28], m-RNN [69], g-LSTM [72] and DeepVS [1] achieved 66.6, 67, 67, and 62.5, respectively. In contrast, our method (Soft) achieved 68.0 and (Hard) achieved 70.3 on BLEU-1, which

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	M	R	C
DeepVS [1]	62.5	45.0	32.1	23.0	19.5	-	66.0
m-RNN [69]	67.0	49.0	35.0	25.0	-	-	-
NIC [28]	66.6	46.1	32.9	24.6	-	-	-
g-LSTM [72]	67	49.1	35.8	26.4	23.9	-	-
LRCN [166]	69.7	51.9	38.0	27.8	22.9	50.8	83.7
Hard-ATT [29]	**71.8**	50.4	35.7	25.0	23.0	-	-
Soft-ATT [29]	70.7	49.2	34.4	24.3	**23.9**	-	-
ConvCap [46]	69.3	51.8	37.4	26.8	23.8	51.1	**85.5**
Ours-Dense (Soft-ATT)	68.0	47.4	32.5	22.9	22.6	**53.0**	74.3
Ours-Dense (Hard-ATT)	70.3	**53.6**	**39.8**	**29.5**	-	-	-

Table 3.1: Performance of our method on MSCOCO dataset. M, R, C stand for METEOR, ROUGE, and CIDEr, respectively (Bold indicates the best result and a dash(-) indicates results are unavailable).

are higher than these methods. The scores for Hard-ATT [29] and Soft-ATT [29] are 71.8 and 70.7, respectively. In terms of score, we are slightly inferior to this popular method. However, our method has superior performance to all the methods on BLEU-2, 3, and 4 metrics. The METEOR metric considers the precision, recall, and the alignment of the matched tokens. The results show that our (Soft) method has better precision and recall accuracy than Karpathy's DeepVS method. ROUGE evaluates the adequacy and fluency of the generated captions, whereas CIDEr focuses grammaticality and saliency. Our method achieved the best result in terms of adequacy, fluency, and saliency.

Qualitative Evaluation Results: We choose some sample images and their ground-truth captions from popular MSCOCO dataset. We generate captions using our method and a baseline attention-based method and show the comparison on the generated captions in Table 3.2. It is seen that the encoder CNN has a big influence on the overall performance of image captioning. Table 3.2 shows some examples of generated captions by different models. It is easy to see that most of the VGGNet-ATT generated image captions are somewhat relevant to the images. However, in some cases, it cannot predict visual attributes properly. For example, in the first image, "horse" is considered an important object even in the ground-truth captions. However, VGGNet-ATT does not include this object in its generated caption. Similarly VGGNet-ATT cannot recognize "dog" in the second, third, and fourth images. In each case, it predicts "cat" instead of 'dog"which makes the generated captions relatively poor. VGGNet-ATT also has problems in distinguishing "man" and "women" as well as "airplanes" and "kites". However, in each case, DenseNet-ATT can recognize the objects properly and include them successfully in the generated captions. We have analyzed a number of images for both VGGNet-ATT generated captions and DenseNet-ATT generated captions. We have found that VGGNet-ATT generates relatively bad captions for many images because it cannot pick the correct information of some specific group of objects. In general, a similar type of objects which do not possess standard appearance often falls in this incorrect recognition. For example, it predicts "man" instead of "woman", "cat" instead of "dog".

Visualization of Attention Probabilities: We visualize attention probabilities generated by VGGNet-ATT and DenseNet-ATT in Figure 3.2 and Figure 3.3, respectively. The image sample is taken from the validation split of MSCOCO dataset. The ground-truth caption for this image is

Image	Captions
	**Ground-Truth Captions:** A man sitting on a horse on a hill. **Generated Captions:** *(VGGNet-ATT):* *A group of cows grazing in a field.* *(Ours DenseNet-ATT (Soft)):* *A horse is standing in a field.* *(Ours DenseNet-ATT (Hard)):* *A man riding a horse down a hill.*
	**Ground-Truth Captions:** A brown and white dog posed on top of a toilet. **Generated Captions:** *(VGGNet-ATT):* *A cat is sitting on a white plate.* *(Ours DenseNet-ATT(Soft)):* *A dog is sitting on a toilet in a bathroom.* *(Ours DenseNet-ATT(Hard)):* *A dog sitting on top of a toilet seat.*
	**Ground-Truth Captions:** A dog that is standing in the grass near a frisbee. **Generated Captions:** *(VGGNet-ATT):* *A cat is laying in a green grass field.* *(Ours DenseNet-ATT(Soft)):* *A dog is sitting in the grass with a frisbee.* *(Ours DenseNet-ATT (Hard)):* *A dog is playing with a frisbee in the grass.*
	**Ground-Truth Captions:** A dog and his owner on the back of a boat. **Generated Captions:** *(VGGNet-ATT):* *A woman holding a piece of pizza.* *(Ours DenseNet-ATT (Soft)):* *A man holding a dog with a surfboard.* *(Ours DenseNet-ATT (Hard)):* *A man sitting on a boat with a dog*
	**Ground-Truth Captions:** Four airplanes are shown flying through a cloudy sky. **Generated Captions:** *(VGGNet-ATT):* *A group of people flying kites in a field.* *(Ours DenseNet-ATT (Soft)):* *A flock of airplanes flying in the sky.* *(Ours DenseNet-ATT (Hard)):* *A plane flying through a cloudy sky.*

Table 3.2: Comparison between our methods and a baseline line method on generated captions. The sample images and their ground-truth captions are collected from MS COCO dataset (Images are best viewed in colour).

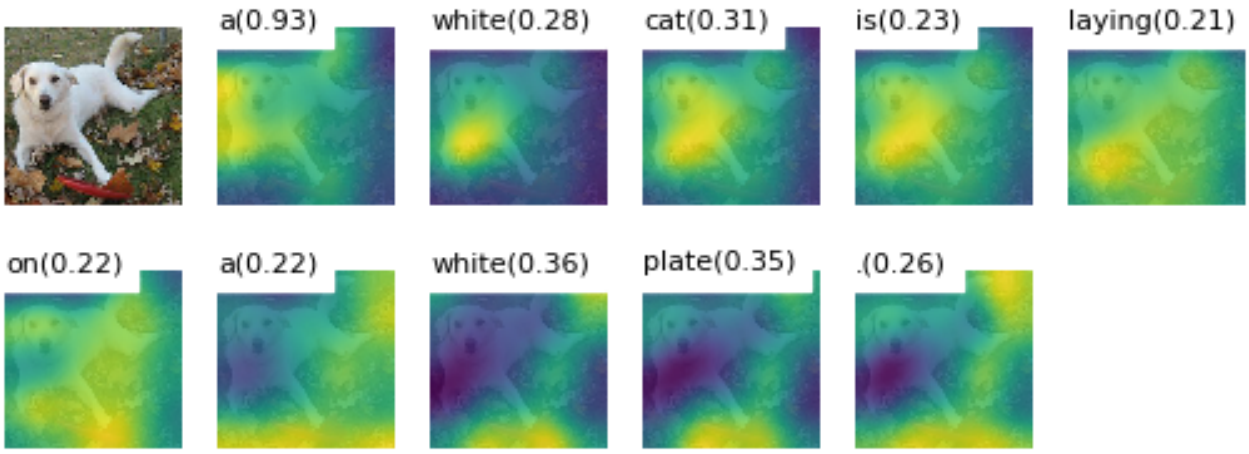


Figure 3.2: Attention visualization generated by VGGNet-ATT.

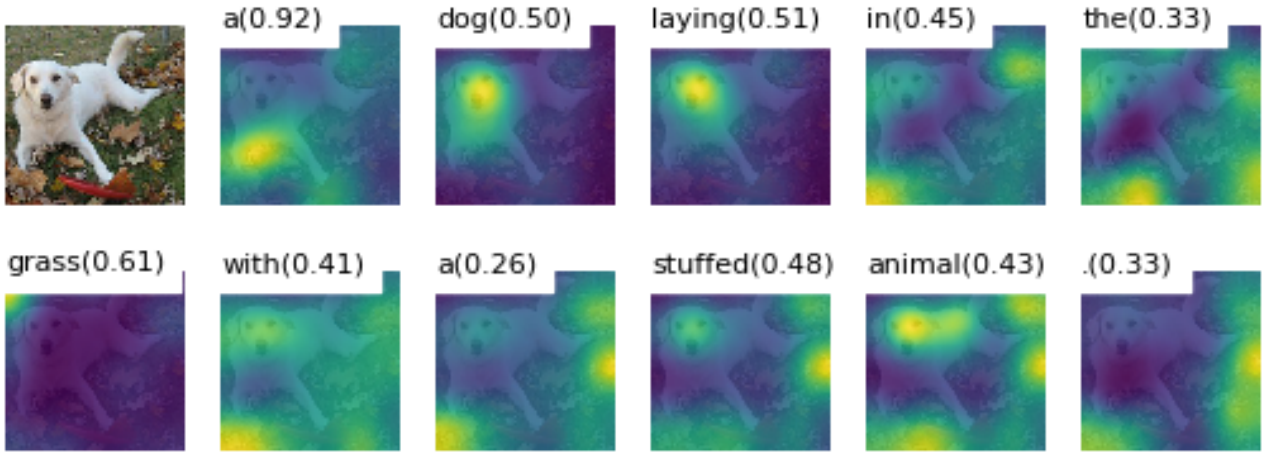


Figure 3.3: Attention visualization generated by DenseNet-ATT.

"Dog sits on the grass with its frisbee". VGGNet-ATT generates a caption for this image as "A white cat is laying on a white plate", and again here it recognizes the "dog" object as "cat". Figure 3.2 shows the attention details for this generated caption. We see each generated word and its relevant regions of the image with its attention probabilities. For example, the fourth block of the first row indicates the attentive colour region for the word "cat". On the otherhand, DenseNet-ATT generates a caption for this image as "A dog laying in the grass with a stuffed animal" where it successfully predicts the "dog" object. The attention details for this caption is given in Figure 3.3. We see that the word "dog" and "laying" mostly focus on the head and the body parts of the image. However, when the network generates the word "in", it shifts its attention to other regions of the image.

3.5 Conclusion

We have proposed an attention-based image captioning method that uses DenseNet features and evaluated its performance on the MSCOCO dataset. DenseNet can extract rich image feature maps and attention mechanism can selectively focus on relevant image features. We have reported our results

on commonly used evaluation metrics such as BLEU, METEOR, ROUGE, and CIDEr, and show that our proposed method achieved better results compared to all the other methods on BLEU-2, 3, and 4 metrics and third best result on BLEU-1. We have also shown the generated captions by our methods and VGGNet-ATT methods. In some cases, our method generates semantically richer captions than VGGNet-ATT. Finally, we have presented the attention visualization details and described how attention shifts from one image region to another based on generated words at each time step.

Chapter 4

Bi-Directional Self Attention for Image Captioning

ABSTRACT

In a typical image captioning pipeline, a Convolutional Neural Network (CNN) is used as the image encoder and Long Short-Term Memory (LSTM) as the language decoder. LSTM with attention mechanism has shown remarkable performance on sequential data including image captioning. LSTM can retain long-range dependency of sequential data. However, it is hard to parallelize the computations of LSTM because of its inherent sequential characteristics. In order to address this issue, recent works have shown benefits in using self-attention, which is highly parallelizable without requiring any temporal dependencies. However, existing techniques apply attention only in one direction to compute the context of the words. We propose an attention mechanism called *Bi-directional Self-Attention (Bi-SAN)* for image captioning. It computes attention both in forward and backward directions. It achieves high performance comparable to state-of-the-art methods.

4.1 Introduction

Image captioning intersects Computer Vision (CV) and Natural Language Processing (NLP). Automatic image captioning is useful to many applications such as developing image search engines with complex natural language queries and helping the people who are visually impaired to understand their surroundings.

Deep learning-based techniques such as CNN [10], Recurrent Neural Networks (RNN), and LSTM [24] have widely been used as they are capable of handling the complexities and challenges of image captioning. Modern image captioning methods typically follow an encoder-decoder framework equipping an attention mechanism. This framework is composed of two principal modules: a CNN as an encoder for image feature extraction and an LSTM as a decoder for caption generation. Popular image

This chapter is published in the proceedings of the International Conference of Digital Image Computing: Techniques and Applications (DICTA). Perth, Australia, 2019, under the title of “Bi-san-cap: bi-directional self-attention for image captioning".

captioning methods such as Show-Attend-and-Tell [29] and Knowing-When-to-Look [191] use this architecture. However, LSTMs ignore the underlying hierarchical structure of a sequence [46]. They require memory storage due to long-term dependencies through a memory cell.

CNNs can also be used in language modeling [32][34]. They can learn the internal hierarchical structures of the sentences and are faster in processing than LSTMs. Inspired by their progresses, some image captioning methods such as [96][46] used CNN as language decoder. However, CNN focuses only on local dependency of a sequence and does not perform well on some tasks such as [192][35].

Due to temporal dependency, LSTMs do not perform parallel computations. CNNs have limitations to learn long-range dependencies of sequences [35]. Self-Attention is another mechanism used for language modeling [35]. It does not require any LSTM/CNN module [35]. It is flexible in modeling long-range as well as local dependencies, and it supports parallel computation. This mechanism achieves state-of-the-art performance on a number of sequential tasks [35][193]. However, it considers attention only in one direction, which does not get rich context for the long sequences[194].

Bi-directional self-attention computes attention both in forward and backward directions to encode the sequential information and feature-level information to handle the variation of contexts around the same word. It applies the forward positional mask to half of the sequence and the backward positional mask to the remaining half. Consequently, it obtains diverse context of the words. It also can perform parallel computation similar to self-attention.

In this paper, we propose an image captioning method, namely Bi-SAN-CAP that has the following key contributions:

- We use bi-directional self-attention in image captioning that uses attention from two directions: forward and backward. It relies solely on the attention to model context dependency and does not need any LSTM or CNN in the decoder part.
- We use two masks, i.e., a forward mask (M^{fw}) and a backward mask (M^{bw}) to compute attentions in both directions.
- We evaluate Bi-SAN-CAP on the popularly used MSCOCO [153] dataset and compare it with LSTM + Attention and CNN + Attention methods.

We organize the rest of the paper as follows: In Section 4.2, we discuss the related work. The architecture and methodology are described in Section 4.3. Experiments and results are discussed in Section 4.4. Section 4.5 concludes the paper.

4.2 Related Work

In the last few years, with the advancements in deep neural network models, automatic image captioning has become a promising research area. Hossain *et al.* [188] present a comprehensive survey of the topic. They group the methods into a number of categories. They include template-based image captioning, retrieval-based image captioning, and novel caption generation. Template-based methods [51][53] use fixed templates with a number of blank slots to generate captions. In these methods,

different objects, attributes, and actions are detected first, and then the blank spaces in the templates are filled. However, templates are predefined and cannot generate variable-length captions.

Captions can also be retrieved from visual space and multimodal space [69][60]. In retrieval-based methods, captions are retrieved from a set of existing captions [1]. These methods produce generalized syntactically correct captions. However, they have limitations in producing image-specific syntactically correct captions [5].

Novel captions can be generated from both visual space and multimodal space [28][29]. A typical method of this category analyzes the visual content of the image first and then generate image captions using a language model. These methods can generate image captions that are semantically more accurate than the aforementioned approaches [5]. Most methods of this category use encoder-decoder architecture to generate image captions [28][195]. In these methods, a vanilla CNN is used as the encoder to extract the image representations and an LSTM is used as decoder to generate captions using this representations. However, these methods have problems in identifying prominent objects of the image.

Attention mechanism can selectively focus on the relevant parts of the image to recognize the prominent objects. A number of encoder-decoder architecture-based image captioning methods [29][72][196][197][198] use attention with LSTM decoder. For example, Ye *et al.* [197] proposed an image captioning method where they use "attentive linear transformation" as attention and LSTM as language decoder. Want *et al.* [2] make use of two LSTMs for capturing past and future context. Bin *et al.* [199] use Bi-directional LSTM and a soft attention mechanism to generate better global representations for videos. Liu *et al.* [200] applied Bi-directional RNN to generate narrative paragraph for photo. Therefore, LSTMs have already been established as a powerful tool for sequence modeling. However, they generate a sequence of hidden states h_t, as a function of the previous hidden state h_{t-1} and the given input for each time step t. This inherently sequential nature excludes parallelization, that becomes critical at longer sequential length. Attention mechanism has compelling advantages in sequence modeling. However, such attention mechanisms are used in conjunction with LSTMs.

CNNs can learn the internal hierarchical structures of the sequence and perform parallel computations. Moreover, optimization in CNNs is easier than LSTMs because the number of non-linearities is fixed in convolutional architectures. Recently, they are used in many sequence tasks [32][34]. Inspired by the success of CNNs in sequence learning tasks, a number of image captioning methods use CNN as the language decoder [96][46][102]. Aneja *et al.* [46] introduced a convolutional language decoder for the task of image captioning. They use a feed-forward network without any recurrent function. It also uses an attention mechanism to leverage spatial image features.

Wang *et al.* [102] proposed another CNN+CNN based image captioning method. This method uses a hierarchical attention module to connect CNN image encoder with the CNN language. However, CNNs are good to extract local dependencies and face difficulties to learn long-range dependencies.

A new attention mechanism called "self-attention" [35] has become popular in many sequence modeling tasks [35][201]. This mechanism is entirely based on attention and does not depend on any LSTM/CNN. Following its merits, it has also been used in image captioning [193][202]. For example, Zhu *et al.* [193] proposed a self-attention-based image captioning method, where the encoder is a CNN as other

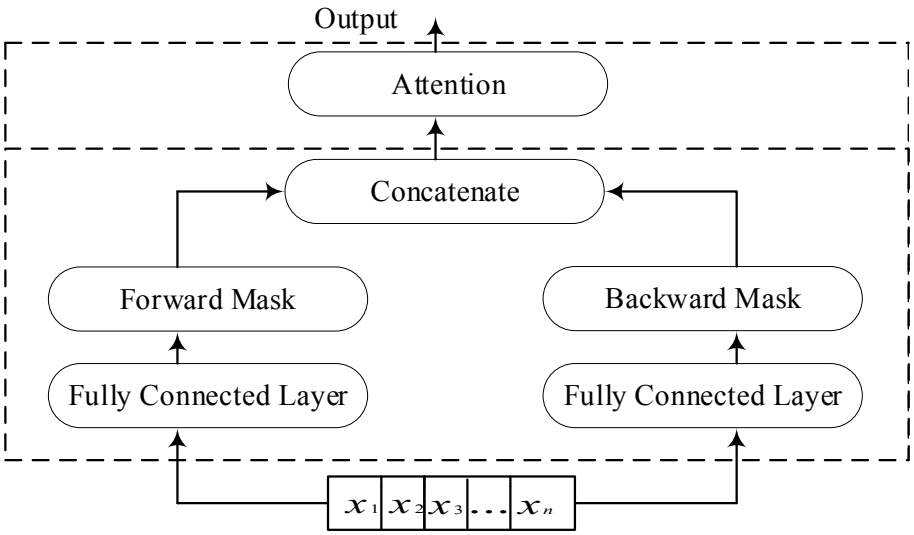


Figure 4.1: Bi-directional self-attention (Bi-SAN) for sequence modeling.

methods but the decoder uses a stacked self-attention with point-wise fully connected layers. This model does not depend on the previous state result in training time. Thus it can perform parallel computation. However, self-attention computes attention only in one direction. Thus, it does not get diverse context of the words.

Bi-directional self-attention (Bi-SAN) computes attention in two different directions. In this paper, we propose a Bi-SAN-based image captioning method, which applies attention in both forward and backward directions to encode the diverse context of the sequence. It also uses the forward and backward masks to encode the temporal order information. The working principle of Bi-SAN is depicted in Figure 4.1. A detailed description is given in Section 4.3.3.

4.3 Model Architecture

The proposed method follows the architecture of the transformer model of Vaswani *et al.* [35]. However, we have modified the encoder side by providing images as input instead of text and allowing to compute bi-directional self-attention. The overall architecture is shown in Figure 4.2. It has two main parts: a CNN encoder and a language decoder. The input of the framework is an image and the output is a language description/caption for that image. The encoder computes the image features from the input image. The decoder takes a text decoding obtained from the training captions and then combines both encodings via input-output mapping. Both the input and the output use Bi-SAN. The input-output mapping involves inter-attention, which allows the architecture to flow information from the encoder to the decoder.

4.3.1 Encoder

CNNs have successfully been used as image encoders in many image captioning methods. In our framework, we use ResNet-101 [16], which is pre-trained on ImageNet [189]. Semantic and global image information can be obtained from the higher layer of a CNN. We use the final convolution layer of the

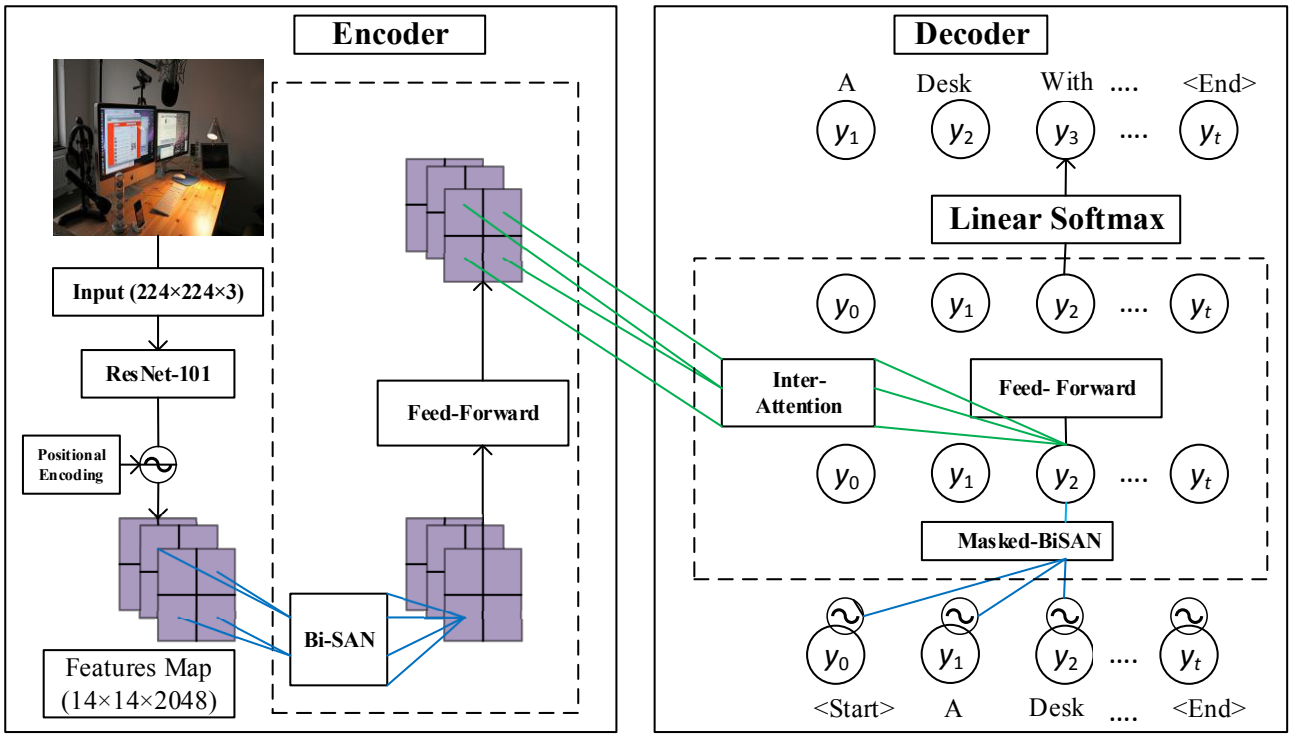


Figure 4.2: The architecture diagram of Bi-Directional self-attention-based Image Captioning.

network to extract the spatial semantic information of the image. We then apply adaptive-pooling, full connected linear and ReLU to obtain a semantic feature map of dimension $d_{model} = 14 \times 14 \times 2048$. Global image feature map can be expressed as:

$$I = I_1, I_2, \ldots, I_{k \times k} \tag{4.1}$$

where $I_i \in \mathbb{R}^{d_{model}}$, $k \times k$ is the number of regions, and I_i represents a region of the image.

The encoder contains six sequentially arranged Bi-SAN layers, each involving multi-head Bi-SAN followed by a feed-forward operation. A detailed description of the attention operation is given in Section 4.3.3. The decoder can access the output of the final Bi-SAN layer during inter-attention.

4.3.2 Decoder

The decoder consists of six layers. Each layer involves "input encoding" and "input-output mapping". Similar to other image captioning methods, we use learned embeddings to convert a word of image captions into d_{model}-dimensional vector. For each caption, the embedding vectors will be a matrix of size $L \times d_{model}$, where L is the length of the caption. Our model does not use any recurrence or convolution. However,the model needs to keep track of the relative or absolute positions of the words in the captions. Positional encodings [34] are used with the words embeddings to make use of the order of the words. The dimension of the positional encodings is the same as the word embeddings d_{model}. The output of the last layer of the decoder is passed to a linear operation followed by softmax to obtain the final output word predictions. For example, during training, "A desk with couple of computer screens on it" is given as input to the decoder of Figure 4.2. At each time step of the decoder, Bi-SAN

is applied to the current word embedding vector of the input and the previous hidden state vector. Then we apply an inter-attention between these results and the encoded image features. The linear softmax classifier predicts the next word of the caption based on this current context.

Masking: A decoder produces output from some hidden states in an autoregressive fashion [45]. It uses the previous output words to generate the next words. Therefore, making use of an encoding of the whole target sequence during training might generate incorrect image captions. In addition to this, producing one word at a time would prevent parallelisation in the decoder. For this reason, the actual target sequence is fed to the decoder during training instead of the previous predicted word. This is a form of a teacher forcing mechanism. In order to still ensure the autoregressive property needed later during inference, attention masking is applied to the bi-directional self-attention weights. Masking restricts the words of the captions to only attend to those at a previous position in the sequence. Moreover, different captions have different lengths. Padding shorter sentences to the same length as the longest one in the batch is the most common solution for this problem. When using padding, we require attention to focus solely on the valid symbols and assign zero weight to pad symbols since they do not carry useful information. The attention-mask handles this issue.

4.3.3 Attention

The attention mechanism selectively focuses on the relevant parts of the information, depending on what is currently being processed. In Bi-SAN, the relevance of a set of values (information) is computed based on some keys and queries. Keys, values, and queries could be anything. The encoder uses input embeddings for its key, values, and queries. On the other hand, the decoder uses the encoder's output for its keys and values and the target sequence embeddings for its queries. In this attention mechanism, the attention weights are the relevance of the encoder hidden states (values) in processing the decoder state (query) and are calculated based on the encoder hidden states (keys) and the decoder hidden state (query). Therefore, an attention function of Bi-SAN can be defined as mapping a query (Q) and a set of key-value pairs (K, V) to an output, where the query, keys, values, and output are all vectors. In terms of equation it can be:

$$A(q, \{(k, v)\}) \xrightarrow[\text{output}]{\text{map as}} \sum_{i=1}^{k} f_c(q, k_i) v_i \tag{4.2}$$

where $q \in Q$ is the query, $k \in K$ is the key, and $v \in V$ is the value; Q, K, and V are vector spaces, f_c is a compatibility function.

The values are weighted and summed to compute the output. In this case, the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. In our model, we use softmax as a compatibility function. The function applies dot-product of the query with all the keys, divided by $\sqrt{d_k}$ to determine the weight of the value. Equation 4.2 can be written as:

$$\mathit{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{4.3}$$

In equation 4.3, $\sqrt{d_k}$ represents the dimensionality of the queries and the keys. d_v represents the

dimensionality of the values.

Multi-Head Attention: Single head computes attention only for one representation of a sequence. For instance, in the sentence "I like cats more than dogs", we might want to consider the fact that the sentence compares two entities. However, we also want to retain the information of the actual entities (cats and dogs) being compared. In our model, we use multi-head attention block considering this issue. This block computes multiple attention weighted sums of the values instead of a single attention. To learn the diverse representations, Multi-Head Attention applies different linear transformations to the values, keys, and queries for each "head" of attention. Thus, the model gets diverse representations of a caption.

The multi-head attention is shown as follows:

$$h_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \tag{4.4}$$

$$H = \text{Concat}(h_1, h_{h2}, \ldots, h_n), \tag{4.5}$$

$$O = HW_h \tag{4.6}$$

where the projections are the parameter metrics $W_i^Q \in \mathbb{R}^{d_{model}} \times d_k$, $W_i^K \in \mathbb{R}^{d_{model}} \times d_k$, and $W_i^V \in \mathbb{R}^{d_{model}} \times d_v$. $Q \in \mathbb{R}^{L \times d_{model}}$, $K \in \mathbb{R}^{L \times d_{model}}$, and $V \in \mathbb{R}^{L \times d_{model}}$ are the inputs of the multi-head attention. *Attention* is the scaled dot-product (multiplicative) attention, *Concat* is the concat function. $h_i \in \mathbb{R}^{L \times d_v}$ is the output of the scaled dot-product attention. n scaled dot-product attentions are concatenated to generate $H \in \mathbb{R}^{L \times (n \times d_v)}$. We use $W_h \in \mathbb{R}^{(n \times d_v) \times d_{model}}$ to project H into the output $O \in \mathbb{R}^{L \times d_{model}}$.

In our framework, we use both image encoding and text encoding in the decoder. At the first layer of the decoder, the keys, values and queries are the same matrices. At the second layer of the decoder, the keys and the values are the matrices generated by the image encoder. The image encoder extracts the spatial image information. The output of the first layer of the decoder is the part of the target caption, which serves as queries for the second decoder layer.

Bi-directional Self-Attention: Masked self-attention applies a mask $M \in \mathbb{R}^{n \times n}$ to the attention alignment score, so that it can allow one-way attention from one token (x_i) to another (x_j). It can be described by the following equation:

$$f(x_i, x_j) = c. \tanh([W^1 x_i + W^2 x_j + b^1]/c) + M_{ij}1 \tag{4.7}$$

where $W^1 \in \mathbb{R}^{d_{feat} \times d_{feat}}$, $W^2 \in \mathbb{R}^{d_q \times d_q}$, b is a bias value, c is a scalar value which is used in the activation function, and 1 is an all-one vector.

In order to model bi-directional order information, the forward mask M^{fw} and the backward mask M^{bw} are respectively substituted into equation 4.7, which results in forward and backward self-attentions. These two attentions are combined by concatenation to compute the bidirectional self-attention. Therefore, we use equations 4.8 and 4.9 for the forward self-attention and the backward self-attention, respectively.

$$f(x_i, x_j) = c. \tanh([W^1 x_i + W^2 x_j + b^1]/c) + M_{ij}^{fw}1 \tag{4.8}$$

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGH-L	CIDEr-D
DeepVS [1]	62.5	45.0	32.1	23.0	19.5	-	66.0
m-RNN [69]	67.0	49.0	35.0	25.0	-	-	-
NIC [28]	66.6	46.1	32.9	24.6	-	-	-
g-LSTM [72]	67	49.1	35.8	26.4	23.9	-	-
Bi-LSTM [2]	67.2	49.2	35.2	24.4	-	-	-
Bi-LSTM-M [203]	68.7	50.9	36.4	25.8	22.9	-	73.9
LRCN [166]	69.7	51.9	38.0	27.8	22.9	50.8	83.7
Hard-ATT [29]	71.8	50.4	35.7	25.0	23.0	-	-
Soft-ATT [29]	70.7	49.2	34.4	24.3	23.9	-	-
ATT-FCN [41]	70.9	53.7	40.2	30.4	23.9	-	-
ConvCap [46]	69.3	51.8	37.4	26.8	23.8	51.1	85.5
GloLocAtt [196]	71.8	54.3	39.5	28.6	**24.2**	52.3	91.2
COMIC [198]	70.6	53.4	39.5	29.2	23.7	51.7	88.1
CT [193]	73.3	**57.0**	43.6	33.3	-	54.8	**108.1**
Bi-SAN-Cap (Ours)	**73.5**	56.7	**44.0**	**34.2**	-	**56.1**	106.7

Table 4.1: Performance of our method on MSCOCO dataset and popularly used evaluation metrics. Bold indicates the best result and a dash(-) indicates results are unavailable.

$$f(x_i, x_j) = c.\tanh([W^1 x_i + W^2 x_j + b^1]/c) + M_{ij}^{bw} 1 \tag{4.9}$$

In forward mask (M^{fw}), there is only attention of later token j to early token i, and vice versa in backward mask (M^{bw}). The two masks are:

$$M_{ij}^{fw} = \begin{cases} 0, & \text{if } i < j \\ -\infty, & \text{otherwise} \end{cases} \tag{4.10}$$

$$M_{ij}^{bw} = \begin{cases} 0, & \text{if } i > j \\ -\infty, & \text{otherwise} \end{cases} \tag{4.11}$$

4.4 Experiments

In this section, we demonstrate the results of our proposed method using the MSCOCO [153] dataset and commonly used evaluation metrics such as BLEU [143], ROUGE [180], and CIDEr [145] for image captioning.

4.4.1 Dataset and Experimental Setup

MSCOCO: Microsoft COCO Dataset [153] is a large and popular dataset for object recognition, segmentation, and image captioning. The dataset consists of 82,783 training and 40,504 validation images. Each image has at least 5 human annotated ground-truth captions. Some of the images have more than 5 annotations. However, we discard the captions above 5 for consistency across other images. We choose MSCOCO dataset because it has much more images and annotations for both training and testing compared to Flickr8K [60] and Flickr30K [152] datasets. We split the images into 3 datasets similar to [1][29]: 82,783 for training, 5,000 for validation, and 5,000 for testing. Our vocabulary contains 20,000 words. The maximum length of our generated caption is 16 words.

Implementation Details: In our framework, we use ResNet-101 with fully connected layers for obtaining image features. We consider the final convolutional layer feature map to compute spatial attention features. The dimension of our feature map is $14 \times 14 \times 2048$. We use a linear layer to obtain a 512 dimensional word embedding in the decoder. The size of CNN and LSTM memory are 196×2018 and 512, respectively. In order to prevent the LSTM network from over-fitting, we add a dropout layer to the output of LSTM. We also apply Adam optimizer [190] with mini-batch size 16 to update the parameters of CNN and LSTM. For the language model part, we set the initial learning rate to be 4×10^{-4}. For CNN, we set the initial learning rate to be 1×10^{-5}, where the momentum and the weight-decay are 0.8 and 0.999, respectively. Then we upsample the word embedding vector via ReLU activation on a fully connected layer, and pass it through a softmax to obtain the output word probabilities $P_{i,w}(y_i|y_{<i}, I)$. Our method was trained for 20 epochs and we evaluate the metrics on the validation dataset, after every epoch, to pick the best model. We use Python (2.7) and Pytorch (0.4.1) deep learning framework to implement our algorithm.

Compared models: We compare our method with several state-of-the-art image captioning methods: DeepVS [1], m-RNN [69], Google NIC [28], g-LSTM [72], Bi-LSTM [2], Bi-LSTM-M [203], LRCN [166], hard-ATT [29], soft-ATT [29], ConvCap [46], GloLocAtt [196], COMIC [198], and CT [193] on MSCOCO dataset and commonly used evaluation metrics in Table 4.1.

Evaluation Metrics: A number of evaluation metrics such as BLEU [143], METEOR [181], ROUGE [180], and CIDEr [145] have widely been used to measure the quality of the generated image captions compared to the ground truth. Each metric applies its own technique for computation and has distinct advantages. In this experiment, we consider BLEU-1, BLEU-2, BLEU-3, BLEU-4, ROUGE, and CIDEr to evaluate our method. For all metrics, higher values indicate better performance.

4.4.2 Result Analysis on MSCOCO Dataset

Table 4.1 shows the results on MSCOCO dataset. The methods use different CNN encoder for image representation. NIC [28] and g-LSTM [72] exploit GoogleNet to extract image features. LRCN [166] utilizes AlexNet to obtain the features. DeepVS [1], m-RNN [69], and Soft/Hard attention [29] use VGGNet to get image-level representation. However, we use ResNet-101 in our method for the task. In terms of the BLEU-1 score, which only considers bigrams, the methods NIC [28], m-RNN [69], g-LSTM [72] and DeepVS [1], COMIC [198], and CT [193] achieved 66.6, 67, 67, and 62.5, 70.6, and 73.3 respectively. In contrast, our method achieves 73.5 on BLEU-1, which is the best result among all the methods. Our method also has superior performance over all the listed methods in BLEU-3 and BLEU-4, and ROUGH-L scores. Our Bi-SAN-Cap produced the second best results on BLEU-2, 56.7, which is slightly inferior to uni-directional self-attention based image captioning model (CT [193]). From the table, it is also seen that, our bi-directional self-attention achieves better results than deep bi-directional LSTM [2] and deep bi-directional LSTM with multi-tasking in all evaluation metrics. Individual text segments are compared in BLEU mteric to measure the quality of the captions. This metric uses n-grams, where $n = 1, 2, 3, 4$. Our model has superior performance in individual word comparisons. The table also shows that our model is better in generating 3 and 4 consecutive correct words than other models. ROUGE evaluates the adequacy and fluency of the generated captions. Our method achieves the best result in terms of adequacy, fluency, and which is salient.

4.5 Conclusion

In this paper, we have presented a Bi-directional self-attention (Bi-SAN) method for image captioning. Bi-SAN applies forward and backward self-attention to obtain the diverse context of the words in captions. Bi-SAN and inter-attention can serve as powerful tools to obtain a rich feature representation. It does not require any recurrence or convolution. Thus, it requires less time in computation similar to self-attention and it also can capture the long-range dependencies of a sequence. We have shown that the proposed Bi-SAN-based image captioning method outperforms other methods on BLEU-1, 3, 4, and ROUGH-L metrics.

Chapter 5

Image Captioning Leveraging Past, Future, and Local Contexts

ABSTRACT

In a typical image captioning pipeline, a Convolutional Neural Network (CNN) is used as an image encoder and Long Short-Term Memory (LSTM) networks are used as a language decoder. LSTM with attention mechanism has shown remarkable performance on sequential tasks including caption generation for images. LSTMs can retain long-term dependency of words in a sentence. However, they ignore the underlying hierarchical structure of a sentence. Therefore, they do not perform well in capturing the local context within a sentence. Beside LSTMs, CNNs can also be used in language modeling. CNNs can learn the internal hierarchical structures of the sentences and hence preserve the local representations. However, they have limitations in capturing the long-term dependencies. In addition, LSTMs can retain information only in forward direction (past context). Bi-directional LSTM (BLSTM) is capable of capturing context both in forward and backward directions, i.e., past and future context, respectively. In this paper, we propose a method where BLSTM is combined with a convolutional structure to extract comprehensive information, namely past, future, and local context information in the caption generation process. For this purpose, a pooling mechanism, called *Attention Pooling*, is used at the pooling stage to harvest the most significant information. We demonstrate our results on MSCOCO dataset using popular evaluation metrics.

5.1 Introduction

Image captioning intersects the research fields of Computer Vision and Natural Language Processing (NLP). Automatic image captioning has a wide range of applications, such as robotic scene understanding, assisting visually impaired people, intelligent human computer interaction, and developing image search engines with complex natural language queries [3], [4].

This chapter is currently under review in the journal of IEEE Transaction on Multimedia, under the title of “Image Captioning Leveraging Past, Future, and Local Contexts".

Deep learning-based techniques such as Convolutional Neural Network (CNN) [10], Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) [24] are capable of handling the complexities and challenges of image captioning. Recently, they have significantly contributed to the advancements of image captioning. In particular, encoder-decoder-based frameworks equipped with attention mechanism have popularly been investigated for image captioning [29], [191]. Such an image captioning framework is composed of two principal modules: a CNN as an encoder to extract visual representations of an input image and an LSTM network as a language decoder to generate captions for that image. However, LSTMs have limitations in extracting the underlying hierarchical structure of a sequence [46]. Therefore, they do not perform well in capturing the local context of the sequence.

CNNs are also used in sequence modeling [32], [34]. They can learn the internal hierarchical structures of sentences. CNNs can independently capture local information contained in every word of a sentence. A number of image captioning methods [96], [46] have already used CNN as a language decoder. However, CNNs focus only on the local dependency of a sentence and do not perform well on a long expression [192][35].

Typical LSTMs work only in forward direction. They can only preserve the past context using the hidden state because they have only seen the information from the past. In contrast, Bi-directional LSTM (BLSTM) compute information in two ways: forward and backward directions. They combine the information using two hidden states and can preserve both past and future contexts. In this paper, we combine BLSTM with a convolutional layer to extract comprehensive information, namely the past, the future, and the local context information of a caption.

A pooling function is used to the feature map obtained by each convolutional filter to reduce the spatial size of the vector representation and so to obtain a fixed length vector. Next, the feature vectors for all the filters are concatenated to form a single feature vector, which is used as an input to the classifier. In this paper, a pooling scheme namely *Attention Pooling* is used to enhance the information extraction capability of the pooling layer.
Overall, in this paper, we propose an image captioning method that has the following key contributions:

- We combine a BLSTM with a convolutional structure as the caption decoder. This combination enables our model to generate captions with comprehensive context information.
- A pooling scheme named *Attention Pooling* is used to preserve the significant information at the pooling state.
- We show the empirical results on MSCOCO dataset which demonstrate that our proposed method achieves comparable performances with the state-of-the-art methods.

We organize the rest of the paper as follows: In Section 5.2, we discuss the related work. The architecture and the methodology of the proposed method are described in Section 5.3. Experiments and results are discussed in Section 5.4. Section 5.5 concludes the paper.

5.2 Related Work

Automatic image captioning has become popular research area with the advent of deep CNN and LSTM-based architectures. A comprehensive survey of the topic has been presented by Hossain *et al.* in [188]. In this survey, the methods of image captioning are grouped into a number of categories such as template-based image captioning, retrieval-based image captioning, and novel caption generation. Template-based methods [51], [53] use manually designed templates with a number of empty slots to generate captions. These methods first identify the information of different objects, their attributes, and their relationships in an image. Then the empty slots of the manually designed templates are aligned with the identified information. Thus, these methods are highly dependent on the predefined templates. Therefore, they cannot accurately express the relationships between the objects in the image in the generated captions.

The retrieval-based methods [69], [60], [1] measure the similarity between an input image and other visually similar images. First, they retrieve the closest matching images based on the similarity. The captions of these retrieved images are regarded as candidate captions. Then the retrieval-based methods choose captions for the input image from these candidate captions. As a result, these methods can generate syntactically correct captions. The retrieved images are visually similar but not exactly same. The combination of different objects, their attributes, and their relationships of these images might be different. Therefore, retrieval-based methods have limitations in generating image-specific novel captions.

Novel captions can be generated from both visual space and multimodal space [28], [29], [204]. A typical method of this category analyzes the visual content of the image and then generates image captions using a language model. These methods can generate image captions that are semantically more accurate than the template-based and the retrieval-based methods. [5]. Most methods of this category use an encoder-decoder architecture to generate image captions[195], [205]. In these methods, a CNN is used as the encoder to extract the image representations and an LSTM network is used as the decoder to generate captions using the image representations.

Kiros *et al.* [63] proposed an encoder-decoder model where an LSTM was used for encoding a sentence and a feed-forward neural language model was used to decode words for generating captions. However, feed-forward neural network-based language models use a fixed length context. Later, in m-RNN model, Mao *et al.* [69] replaced the feed-forward neural language model with RNNs. Neural Image Captioning (NIC) and Long-term Recurrent Convolutional Neural Network (LRCN) methods introduced by Vinyals *et al.*[28] and Donahuel *et al.* [166] used LSTM to learn text context. However, NIC feeds visual information into LSTM only at first time step. In contrast, m-RNN and LRCN feed this visual information to LSTM at each time step. Xu *et al.* used LSTM with attention as the language decoder. This method can selectively focus on the relevant objects in an image based on the generated words by LSTM at each time step. Recently, Huang *et al.* [206] proposed a subsequent attribute predictor LSTM (SAP-LSTM) to predict the next word for caption generation. The SAP module dynamically predicts the most relevant attributes of the objects of an image at each time step of LSTM.

Recently, CNNs are used in many sequence tasks [32], [34]. Inspired by the success of CNNs in sequence learning tasks, a number of image captioning methods use CNN as the language decoder [96], [46],

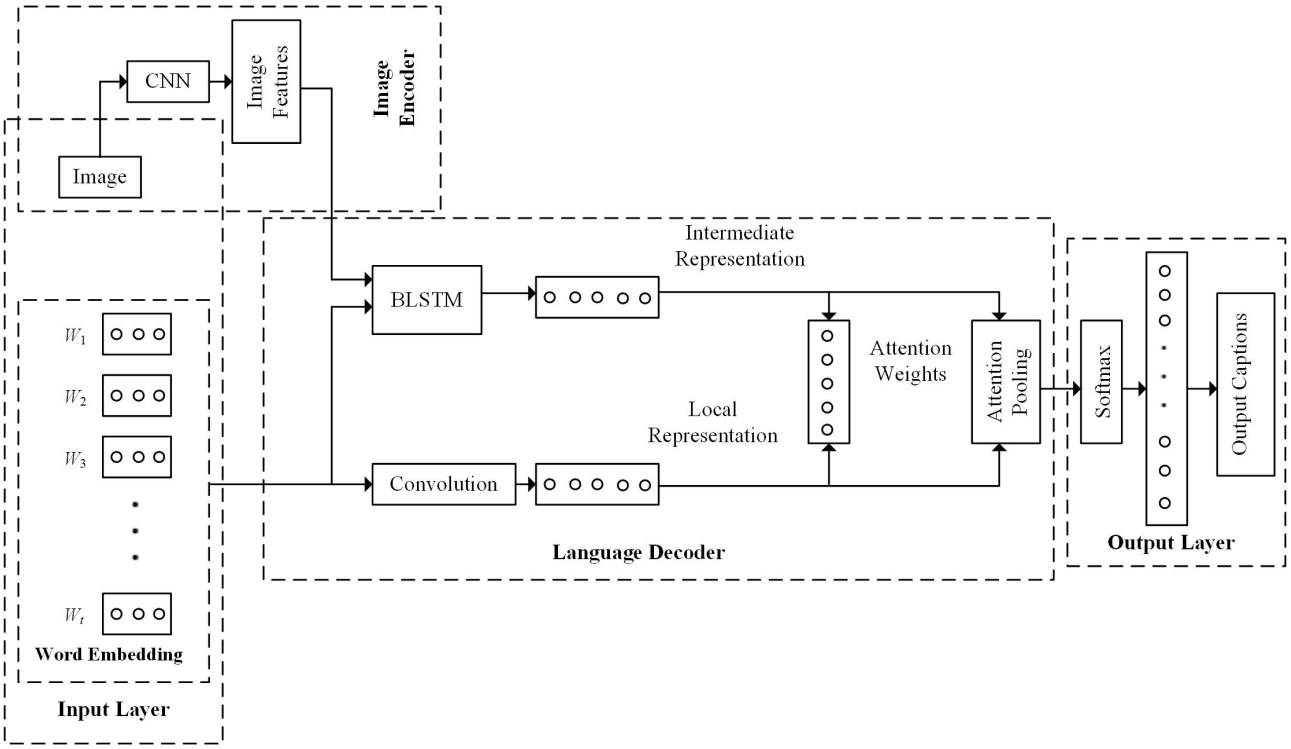


Figure 5.1: The architecture of our proposed method: a CNN is used as Image Encoder. In language decoder, a BLSTM combined with a Convolution structure is used to learn past, future, and local context for generating image captions. Finally, an attention pooling technique is applied to retain the most significant information at the pooling stage. Figures are best viewed in color.

[102]. Aneja *et al.* [46] introduced a convolutional language decoder for the task of image captioning. They use a feed-forward network without any recurrent function. It also uses an attention mechanism to leverage spatial image features.

Wang *et al.* [102] proposed another CNN+CNN based image captioning method. This method uses a hierarchical attention module to connect CNN image encoder with the CNN language decoder. However, CNNs are good to extract local dependencies and face difficulties to learn long-range dependencies.

Wang *et al.* [2] proposed a BLSTM based image captioning method. This method is capable of using past and future context information to generate image captions.

Pooling functions can reduce the number of parameters of a model. They alleviate the problem of over-fitting. A number of pooling functions such as max pooling [20], average pooling [21], and k-max pooling [11] have commonly been used in the model. These functions have a tendency to discard context information to some extent.

In this paper, we propose an image captioning method, where a BLSTM is combined with a CNN structure in the decoding module. We apply a pooling function called *Attention Pooling* that can preserve the most significant information at the pooling stage. Er *et al.* [207] applied this technique for sentence modeling. However, our pipeline is different from [207]. While the input for their BLSTM is only word embedding, our BLSTM takes two inputs: the image features and the word embedding. This additional component of BLSTM guides our method to generate description from images. We use a CNN as image encoder in our method to extract the image features. The two different networks

namely, a BLSTM and a convolutional structure together with the *Attention Pooling* in the decoder can extract past, future, and local context information in generating a description of an image.

5.3 Model Architecture

The overall architecture of our proposed method is shown in Figure 5.1. It has two main modules: a CNN-based Image Encoder and a Language Decoder. The input of the framework is an image and the output is a language description/caption for that image. The encoder computes the image features from the input image. These image features are then fed into BLSTM. The language decoder, which has two main components: a BLSTM and a convolutional layer. Caption words are represented by a technique called word embedding. Both BLSTM and convolutional layers take this word embedding as input. The intermediate representations generated by BLSTM and the local representations generated by convolutional layer are combined to compute the attention weights. Finally, an attention pooling layer followed by a softmax layer is used to generate the captions.

5.3.1 Image Encoder

The goal of an image captioning module is to generate a description of an image. Performance of such a module depends on two principles: (i) How well it recognizes the different objects, their attributes, and their relationships in image. (ii) How accurately it devises them into a natural language sentence.

Traditional convolutional networks with L layers have L connections whereas DenseNet [17] has $L(L+1)/2$ direct connections. As a result, the feature-maps of all preceding layers are used as inputs to the current layer, and its own feature-maps are used as inputs into all subsequent layers. The transformation function for DenseNet is:

$$I_l = H_l([I_0, I_1, \ldots, I_{l-1}]) \tag{5.1}$$

where $[I_0, I_1, \ldots, I_{l-1}]$ refers to the concatenation of the feature-maps generated in layers $0, 1, \ldots, l-1$ and $H_l(.)$ is a composite function. In our experiments, we use DenseNet as the image encoder which is pre-trained on ImageNet [189].

5.3.2 Language Decoder

We used a convolutional structure with a BLSTM to decode words for generating description of an image. Convolutional filters are used to perform convolution on the input word embedding matrix. Therefore, we get a local representations of the text features. These filters are capable of capturing the local context of every word in a sentence.

BLSTM [31], [24] computes information using both the forward hidden layer and the backward hidden layer. Therefore, it can explore the past and the future context information of a sequence. In our experiments, we use the representation from an intermediate layer of BLSTM to compute the attention weights. The attention weights are obtained by comparing the local representations position by position with an the intermediate representation generated by the BLSTM.

Finally, caption representations of all distinct convolutional filters are concatenated into the final feature vector which is fed into a top-level softmax classifier. The intermediate representation generated by the BLSTM, the local representation from the convolutinal layer, and the attention weights are used as input to the pooling layer. This pooling strategy termed as *Attention Pooling* is used to retain the most significant information of the historical, future, and local context of the sequence. Then, a softmax classifier is used to predict the next word in generating captions.

The salient components of the language model, namely *Attention Pooling* and the combination of BLSTM with convolutional structure are described in Section 5.3.2. We also describe other necessary components, namely word embedding, and convolution method, in the same Section, respectively to make the description complete and comprehensive.

Word Embedding

The words usually need to be represented by word vectors such as one-hot vectors and word embeddings before feeding into machine learning systems. One-hot vector has shown good learning performance in machine learning applications such as document classification [208]. However, one-hot vectors are high-dimensional and sparse. Consequently, they are semantically and computationally less efficient. In contrast, word embeddings (e.g., Word2vec, GloVe) are low-dimensional and dense. They are represented as continuous vectors. Moreover, in word embedding, words with similar meanings end up with a similar vector representation. Therefore, word embeddings with all these attributes are powerful and efficient for machine learning algorithms. A word can be represented by a dense vector as follows:

$$X = Lw \tag{5.2}$$

where $w \in \mathbb{R}^V$ is a one-hot vector where the position that the word appears is one while the other positions are zeros, $L \in \mathbb{R}^{d \times v}$ is a word-representation matrix, in which the ith column is the vector representation of the ith word in the vocabulary, V is the vocabulary size.

We use pre-trained word embeddings to make better use of syntactic and semantic associations of words. Word2vec [173] and GloVe [173] are two widely used pre-trained word embedding matrices. Several works [209], [210], [211] have demonstrated that Word2vec performs better than GloVe. For example, Levy *et al.* [209] used both Word2vec and GloVe to compute the word similarity. They reported 79.3% and 72.5% on Word2vec and GloVe, respectively, in computing word similarity. In this paper, we use word2vec to represent the words of the captions. The model is trained on 100 billion words from the Google News to maximize the average log probability as follows:

$$\frac{1}{T}\sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} = logp(X_{t+j}|X_t) \tag{5.3}$$

where c is the context window size. The values of word vectors are included in the parameters, which are optimized during the training procedure.

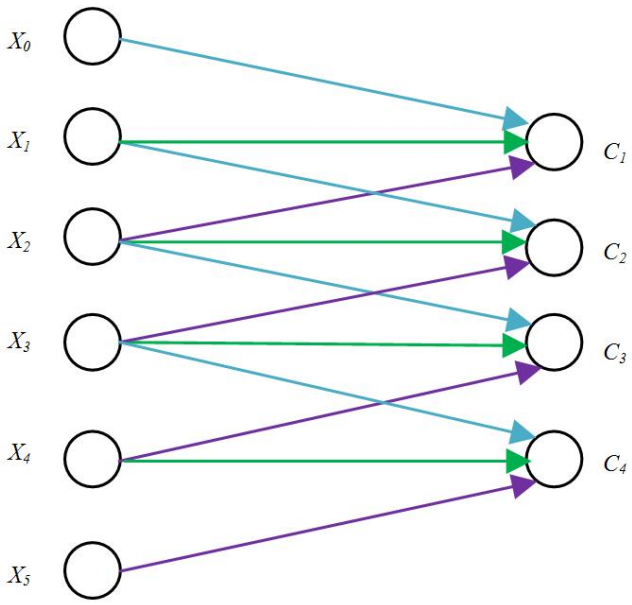


Figure 5.2: An illustration of a convolutional graph. The weights are shared across windows of the filter. The arrows with the same colour represent the same weight values. The $X_{i:i+m-1}$ is a concatenation vector. The vector represents a window of m words starting from the ith word, obtaining features for the window of words in the corresponding feature maps. Each element C_i represents the local context of the corresponding position. Figures are best viewed in color.

Convolution Layer

Convolutional layers are the major building blocks of a CNN. These convolution layers apply filters to perform convolution operation to input data. Generally in computer vision it is accepted that deep architectures with multiple convolutional layers can learn features at various levels of abstraction to achieve good performance [212]. However, only one convolutional layer can achieve state-of-the-art or comparable performance on NLP related tasks [213]. In some cases, one may obtain marginal or even decreasing performance because of over-fitting with increasing number of convolutional layers. Furthermore, if the depth of the network exceeds than a certain level, the computation complexity increases quickly. In this paper, we also use one convolution layer for the convolutional structure of the decoder. This type of convolution is shown in Figure 5.2.

We conduct the convolution operation on word vectors in one dimension between filters $w_c \in \mathbb{R}^{md \times k}$ and a concatenation vector $X_{i:i+m-1}$. We calculate features for the window of the words in the corresponding feature maps. Thus we obtain the feature representation for total m words. The term d is the dimension of word embedding. The length of all the captions are not same. In our method, we suppose the length of the caption is T. We set the border mode of convolution is same for both the input sequence and generated output. We use zero-paddings to guarantee the same length for each sequence. In order to improve the model's learning capability we use multiple filters with different initialized weights. Convolution operation with k filters can be written as:

$$C_i = g(W_c^T X_{i:i+m-1} + b_c) \in \mathbb{R}^k \tag{5.4}$$

where $X_i \in \mathbb{R}^d$, the term b_c is a bias vector and $g(\cdot)$ is a nonlinear activation function. The ReLU has

become a standard nonlinear activation function of CNN recently because it can improve the learning dynamics of the network and significantly reduce the number of iterations required for convergence in deep learning networks. ReLU has been used effectively with CNN as a standard nonlinear activation function. We use LeakyReLU [214] which is another version of ReLU that can further improve the learning efficiency of the model. If the length of a caption is T, the feature maps of the convolutional layer can be represented as:

$$C = [C_1, C_2, \ldots, C_T] \in \mathbb{R}^{k \times T} \tag{5.5}$$

We obtain the local representation of each caption from the output of the convolutional layer. Each element C_i represents the local context of the corresponding position.

Attention Pooling

A pooling layer is an important building block of a CNN. This layer uses a number of pooling function such as max pooling, average pooling, and k-max pooling to down sample the feature maps by summarizing the presence of features in feature map. Pooling is also used to reduce the number of parameters and computations in the network. All these existing pooling functions discard intensity information of features to some extent. In this paper, we use a pooling mechanism called *Attention Pooling* that can preserve the most significant information at the pooling stage.

BLSTM is a variant of Recurrent Neural Network (RNN). It is also able to learn both the past and the future context information of a sequence. It has already been successfully used in sequence modeling [31], [2], [215]. In our proposed method, we use an intermediate representation of BLSTM, which is denoted as $\tilde{s}$. Then we extract the local representation using a convolutional layer. This representation is presented by c_i. Then we map and compare the local representation with the intermediate representation to calculate the attention weights. We compute the similarity between the local representation and each intermediate representation. The higher the similarity, the bigger attention weight is assigned. The attention weight of each word determines how much significance each word has in generating a semantically meaningful caption. Thus attention pooling can preserve the most significant information at the pooling stage. The attention weights are computed as:

$$\alpha_i = \frac{\exp(e_i)}{\sum_{i=1}^{T} \exp(e_i)} \tag{5.6}$$

where

$$e_i = sim(c_i, \tilde{s}) \tag{5.7}$$

The term α_i is a scalar and the function $sim(\cdot)$ is used to measure the similarity between its two inputs. Cosine similarity is used in our method. Therefore, the local context of CNN layer together with the intermediate representation of BLSTM triggers the model to generate caption with historical, local, and future context information. The final caption is represented by:

$$S = \sum_{i=1}^{T} \alpha_i c_i \in \mathbb{R}^k \tag{5.8}$$

5.4 Experiments

In this section, we demonstrate the results of our proposed method using the MSCOCO [153] dataset and commonly used evaluation metrics such as BLEU [143], ROUGE [180], and CIDEr [145] for image captioning.

5.4.1 Dataset and Experimental Setup

MSCOCO

Microsoft COCO Dataset [153] is a large and popular dataset for object recognition, segmentation, and image captioning. The dataset consists of 82,783 training and 40,504 validation images. Each image has at least 5 human annotated ground-truth captions. Some of the images have more than 5 annotations. However, we discard the captions above 5 for consistency across other images. We choose MSCOCO dataset because it has much more images and annotations for both training and testing compared to Flickr8K [60] and Flickr30K [152] datasets and it is more challenging. We split the images into 3 datasets similar to [1], [29]: 82,783 for training, 5,000 for validation, and 5,000 for testing. Our vocabulary contains 20,000 words. The maximum length of our generated caption is 16 words.

Implementation Details

In our framework, we use DenseNet121 [17] with fully connected layers for obtaining image features. The DenseNet121 is pre-trained on ImageNet dataset [189]. We apply the fc7 feature map to compute the features. The dimension of the feature map from DenseNet is 1×1024. The size of the hidden layer in the prediction module is 1024. BLSTM and CNN structure are used to extract the text features. The filter window size for this CNN structure and the number of features maps from both BLSTM and CNN structure have large effects on the performance of the model. We choose the optimal filter window size 3. When the number of feature maps is small, the accuracy of the model can be small. However, as the size increases, the performance does not improve too much and can even deteriorate because of the problem of over-fitting. Furthermore, the complexity of the model increases quickly with the increasing number of feature maps. Therefore, we set the the number of feature maps as 200. We apply dropout, learning rate to 0.001 and use a linear layer to obtain a 512-dimensional word embedding. We also apply Adam optimizer [190] with mini-batch size 16 to train the model. Then we upsample the word embedding vector via ReLU activation on the fully connected layer, and pass it through a softmax to obtain the output word probabilities $P_{i,w}(y_i|y_{<i}, I)$. Our method was trained for 20 epochs and we evaluate the metrics on the validation dataset, after every epoch, to pick the best model. The model was implemented in Tensorflow 1.8.

Compared models

We compare our method with several state-of-the-art image captioning methods: m-RNN-2014 [69], DeepVS-2015 [1], Google NIC-2015 [28], LRCN-2015 [166], Hard-ATT-2015 [29], Soft-ATT-2015 [29], Bi-LSTM-2016 [2], ATT-FCN-2016 [41], GloLocAtt-2017 [196], SCA-CNN-2017 [216], Bi-LSTM-M-2018 [203], CNN-CNN-2018 [102], ConvCap-2018 [46], Paying-2018 [217], Scene-2019 [218], Image-2019 [219],

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGH-L	CIDEr
m-RNN-2014 [69]	67.0	49.0	35.0	25.0	-	-	-
DeepVS-2015 [1]	62.5	45.0	32.1	23.0	19.5	-	66.0
NIC-2015 [28]	66.6	46.1	32.9	24.6	-	-	-
LRCN-2015 [166]	69.7	51.9	38.0	27.8	22.9	50.8	83.7
Hard-ATT-2015 [29]	71.8	50.4	35.7	25.0	23.0	-	-
Soft-ATT-2015 [29]	70.7	49.2	34.4	24.3	23.9	-	-
Bi-LSTM-2016 [2]	67.2	49.2	35.2	24.4	-	-	-
ATT-FCN-2016 [41]	70.9	53.7	40.2	30.4	23.9	-	-
GloLocAtt-2017 [196]	71.8	54.3	39.5	28.6	24.2	52.3	91.2
SCA-CNN-2017 [216]	71.9	54.8	41.1	31.1	25.0	-	-
Bi-LSTM-M-2018 [203]	68.7	50.9	36.4	25.8	22.9	-	73.9
CNN-CNN-2018 [102]	68.8	51.3	37.0	26.7	23.4	51.0	84.4
ConvCap-2018 [46]	71.0	53.7	39.1	28.4	24.4	51.9	89.9
Paying-2018 [217]	70.8	53.6	39.1	28.4	24.8	52.1	89.8
Scene-2019 [218]	67.9	49.4	34.7	24.3	22.2	48.8	75.4
Image-2019 [219]	68.5	50.2	36.8	27.3	23.2	50.3	85.2
COMIC-2019 [198]	70.6	53.4	39.5	29.2	23.7	51.7	88.1
Learning-2020 [3]	***75.9***	***60.3***	***46.5***	***35.8***	***27.8***	***56.4***	***109.4***
Leveraging-2020 [220]	73.1	56.1	43.1	32.6	25.0	-	98.4
Stimulus-2020 [221]	74.8	52.5	36.5	23.5	23.5	50.5	104.1
Ours_(BLSTM+CNN)	72.1	**56.3**	**43.2**	**33.0**	-	**54.3**	**106.7**

Table 5.1: Performance of our method on MSCOCO dataset and popularly used evaluation metrics. Bold and italic, bold, and a dash(-) indicate the best result, the second best result, and the results are unavailable, respectively.

COMIC-2019 [198], Learning-2020 [3], Leveraging-2020 [220], and Stimulus-2020 [221] on MSCOCO dataset and commonly used evaluation metrics in Table 5.1.

Evaluation Metrics:

A number of evaluation metrics such as BLEU [143], METEOR [181], ROUGE [180], and CIDEr [145] have widely been used to measure the quality of the generated image captions compared to the ground truth. Each metric applies its own technique for computation and has distinct advantages. In this experiment, we consider BLEU-1, BLEU-2, BLEU-3, BLEU-4, ROUGE, and CIDEr to evaluate our method. For all metrics, higher values indicate better performance.

5.4.2 Analysis of Result

We discuss and analyze both qualitative and quantitative results of the generated captions.

Quantitative Analysis

Table 5.1 shows the performance of the compared methods on MSCOCO dataset and commonly used evaluation metrics. The methods CNN-CNN-2018 [102] and ConvCap-2018 [46] use CNN as a language decoder whereas Bi-LSTM-2016 [2] and Bi-LSTM-M-2018 [203] methods use BLSTM for the same task. All other methods except our method in the table use LSTM for language representation. Our method use a combined representations of a BLSTM and a CNN layer for this purpose. Overall, the results across the six evaluation metrics indicate that our proposed method achieve comparable performances with the state-of-the-art methods. In particular, our method can achieve 72.1, 56.3, 43.2, 33.0, 54.3, and 106.7 in BLEU-1, 2, 3, 4, ROUGH-L, and CIDEr-D respectively, making the superior performance over the methods which use either CNN or BLSTM as language decoder. From the table, it is also

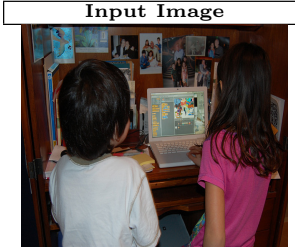
Input Image	Output Captions
	**Ground-Truth Captions:** 1. One boy and one girl in front of a laptop. 2. The children are enjoying an activity at home on the laptop computer. 3. Couple of children on lap top playing a game . 4. Two children at a desk with a laptop. 5. A bunch of little kids that are sitting at a laptop. **Generated Captions:** **LSTM:** *A woman in a red shirt is holding a remote.* **CNN:** *A woman in a living room with a laptop.* **BLSTM:** *A woman sitting on a desk with a laptop.* **Ours_(BLSTM+CNN):** *One boy and one girl at home on the laptop.*
	**Ground-Truth Captions:** 1. A green bird with purple eyes stands on a yellow perch. 2. A bird perched on a banana with trees in the background . 3. A large green bird is standing on peeled bananas . 4. A close up of a bird standing on a banana. 5. A green and white bird standing on a banana. **Generated Captions:** **LSTM:** *A white bird sitting on a white table .* **CNN:** *A white bird sitting on a banana.* **BLSTM:** *A large green bird standing on bananas.* **Ours_(BLSTM+CNN):** *A green bird standing on peeled bananas in a background.*
	**Ground-Truth Captions:** 1. A yellow bus and a blue bus drive next to each other in the city. 2. A couple of city buses ride on a city street . 3. Yellow and blue double decker buses traveling side by side. 4. Two buses driving down a curvy street next to a building. 5. Two double high buses that are sitting in the street. **Generated Captions:** **LSTM:** *A bus driving down a street in front of a building.* **CNN:** *A bus driving down a street next to a bus.* **BLSTM:** *A double decker bus driving down a city street next to a bus.* **Ours_(BLSTM+CNN):** *A yellow and a blue bus driving down a street in front of a building.*

Table 5.2: Comparison of our method with the state-of-the-art-methods on generated captions. The input image samples and their ground-truth captions are collected from the MS COCO dataset. 'LSTM' , 'CNN', and 'BLSTM' means the language decoder is based on LSTM, CNN, and BLSTM, respectively. Images are best viewed in color.

seen that our method perform much better in all the evaluation metrics than most of the methods which use LSTM only as a language decoder.

The methods Learning-2020 [3], Leveraging-2020 [220], and Stimulus-2020 [221] achieve 75.9, 73.1, and 74.8, respectively on BLEU-1 metric while our method achieves 72.1, which is slightly inferior to these aforementioned methods. The table also shows that our proposed method achieves 56.3, 43.2, 33.0, 54.3, and 106.7 in BLEU-2, 3, 4, ROUGH, and CIDEr metrics, respectively. These results indicate that our method can produce the second best results over all the listed methods on all the evaluation metrics except BLEU-1.

Qualitative Analysis

We choose some sample images and their ground-truth captions from MSCOCO dataset. We have illustrated the captions generated by our method and three other different methods for these images in Table 5.2. The three methods we have presented in the Table use either a typical LSTM, CNN, or BLSTM as language decoder. We use both a BLSTM and a CNN structure in our method for language representation. Then we have analyzed and compared the performances of these generated captions. It can be seen from the Table 5.2 that our method can generate semantically more meaningful and superior captions to those generated by typical LSTM, CNN or BLSTM. In the first example of this table, all the three methods cannot recognize "the boy and the girl" properly whereas our method can recognize them successfully. Although CNN-based and BLSTM-based methods recognize

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGH-L	CIDEr
Average Pooling	68.2	50.6	36.4	26.0	50.1	76.3
Max Pooling	68.9	52.3	37.8	26.7	51.6	79.1
Attention Pooling	**72.1**	**56.3**	**43.2**	**33.0**	**54.3**	**106.7**

Table 5.3: An ablation performance summary of our method using different pooling strategies. Bold indicates the best result.

the object “laptop", LSTM-based method cannot recognize it. However, our method can successfully generate not only the word “laptop" but also include an additional relevant word “home" in the caption. Similarly, LSTM and CNN-based methods incorrectly generate “white bird" instead of “green bird" in the second example. LSTM-based method also misrecognizes the object “table". Although CNN-based and BLSTM-based methods can recognize “banana" properly, they do not include anything about “background". However, our method can successfully recognize “green bird", “bananas" including the word “background" in generating more meaningful and accurate caption. It can also be seen from the Table 5.2 that our method can pick the relevant words appropriately for example three. The example shows that our method can recognize both of the buses including their colors. It also includes other relevant words such as “driving", “street", and “building", which make the generated caption superior to those generated by other methods mentioned in the table.

5.4.3 Ablation Studies

We conducted ablation experiments to compare the different pooling strategies used in our method at the pooling stage. **Average pooling** [21] returns the average value of the feature map. It is identified that the sharp features may not be chosen when average pooling is used [222]. **Max pooling** [223] selects the highest value from the feature map. Thus it can capture the most relevant feature. However, it loses position and intensity information of the features because only the maximum value is used at the pooling stage. In **Attention pooling**, first, the local representations generated by convolutional layer are compared with the intermediate representations BLSTM to calculate the attention weights. The higher the similarity between the intermediate representation and each local representation, the bigger attention weight is assigned to the local representation of each word in pooling layer. Then a weighted sum of all the word annotations are used to compute the final predictions. Thus attention pooling can encode the richer information to generate the captions.

Table 5.3 reports the performance of different pooling techniques used in our method. From the results, we can obtain the following observations. The **Attention pooling** based method achieves 72.1, 56.3, 43.2, 33.0, 54.3, 106.7 in BLEU-1, 2, 3, 4, ROUGH, and CIDEr metrics, respectively whereas the **Average pooling** based method achieves 68.2, 50.6, 36.4, 26.0, 50.1, and 76.3 in these metrics. These scores for **Max pooling** based method are 68.9, 52.3, 37.8, 26.7, 51.6, and 79.1. It is observed that the results achieved by the **Attention pooling** based method are the best in Table 5.3. It indicates that the **Attention pooling** based method outperforms the **Average pooling** and **Max pooling** based methods significantly in terms of all the evaluation metrics. It can also be implied that the **Attention pooling** can preserve significant information at the pooling stage which drives our method to achieve high scores. It is also observed that though the **Max pooling** achieves lower scores than

the **Attention pooling**, it performs better than the **Average pooling** over all the metrics.

5.5 Conclusion

LSTM can preserve only historical context information of a sequence while BLSTM can capture both the historical and the future context. They have popularly been used for sequence modeling. However, they have limitations in capturing the local context of a sequence. CNNs can also be used in sequence modeling. They are capable to capture the local context of a sequence. Therefore, we have used both a BLSTM and a CNN layer as language decoder in our image captioning model. We have compared the intermediate representation generated by BLSTM with the local representation from CNN layer to compute the attention weights. Then we have applied an attention pooling mechanism at the pooling stage. Finally a softmax layer is used to predict the words for generating captions. We have analysed the results using both quantitative and qualitative analysis. Then we have compared the results with the state-of-the-art methods. From the quantitative results it can be seen that our method achieve comparable performances with the state-of-the-art methods. For example, our method achieve the second best results over all the listed methods in all the evaluation metrics except BLEU-1. It can also be seen from qualitative results that our method can generate semantically meaningful and superior captions to those generated by typical LSTM, CNN or BLSTM.

Chapter 6

Text to Image Synthesis for Improved Image Captioning

ABSTRACT

Generating textual descriptions of images has been an important topic in computer vision and natural language processing. A number of techniques based on deep learning have been proposed on this topic. These techniques use human-annotated images for training and testing the models. These models require a large number of training data to perform at their full potential. Collecting human generated images with associative captions is expensive and time-consuming. In this paper, we propose an image captioning method that uses both real and synthetic data for training and testing the model. We use a Generative Adversarial Network (GAN) based text to image generator to generate synthetic images. We use an attention-based image captioning method to generate the captions. To the best of our knowledge, there is no method available in image captioning which uses GAN for image synthesis. We demonstrate the results of our models using both qualitative and quantitative analyse on popularly used evaluation metrics. We show that our experimental results achieve two fold benefits of our proposed work: i) it demonstrates the effectiveness of image captioning for synthetic images, and ii) it further improves the quality of the generated captions for real images, understandably because we use additional images for training.

6.1 Introduction

Image captioning is the task of providing a natural language description of the content in an image and lies at the intersection of computer vision and Natural Language Processing (NLP) [224]. Automatic image captioning is useful to many applications, such as developing image search engines with complex natural language queries and helping the visually impaired people to understand their surroundings. Hence, image captioning has been an active research area. The advent of new convolutional neural networks and object detection architectures have contributed enormously to improving image captioning.

This chapter is currently under revision from the journal of IEEE Access, under the title of “Text to Image Synthesis for Improved Image Captioning".

Moreover, sophisticated sequential models, such as attention-based recurrent neural networks, have also been presented for accurate image caption generation.

Inspired by neural machine translation, most modern deep learning-based image captioning methods use an encoder-decoder framework. In this framework, an encoder is used to encode an intermediate representation of the information contained within the image. A decoder is used to decode this information into a descriptive text sequence. Thus this framework is composed of two principal modules: a Convolutional Neural Network (CNN)[10][225] as an encoder for image feature extraction and a Long Short-Term Memory (LSTM) model [24] as a language decoder for caption generation.

Different CNNs such as AlexNet [14], VGGNet [13], ResNet [16], and DenseNet [17] have their own strengths and weaknesses. It is generally accepted that the deeper the network is, the more relevant are the learned features [16]. However, if the depth of the network exceeds a threshold, one may obtain the opposite effect, i.e., a decline in performance. There are two main reasons behind this fact: (i) The vanishing-gradient problem: when the input or the gradient passes through many layers, it can vanish or gets "washed out" by the time it reaches the end of the network, and (ii) the degradation problem. This problem has been addressed in the literature by using residual learning mechanisms such as ResNet [17]. However, the element-wise addition used in the identity mapping in ResNet is computationally expensive during training. In contrast, with DenseNet, each layer has connections with every other layer in the network in a feed-forward manner. The network reuses the feature-maps and uses concatenation for various operations instead of addition. Therefore, it can reduce the number of parameters and it can be memory efficient. Moreover, since each layer of DenseNet receives feature maps from all previous layers, it gets diversified features and tends to have rich patterns. In this paper, we use DenseNet as an encoder to extract image features.

However, encoder-decoder based methods focus only on the factual description of an image. They lose the information of the relevant objects in the scene. Visual attention mechanisms can selectively focus on the relevant parts of the image for a period of time, similar to the human visual system. Simultaneously, they can discard irrelevant information. Several methods [205][226] use attention-based techniques and can describe the relevant parts of the image successfully. All of these methods use the three most common datasets: Microsoft COCO (MSCOCO) [153], Flickr30k [152], and Flickr8k [60]. The images of all these datasets are human-annotated. However, these deep learning-based methods require a large amount of labeled data in order for them to perform at their very best. Moreover, the manual generation of (additional) data is expensive and time-consuming [227].

Nowadays a lot contents including images are generated automatically, e.g., for news, illustration, artwork, promotion, as well as for human computer interaction and augmented reality. Such synthetic data can be effectively used in machine learning techniques, where there is a scarcity of labelled data.

Application such as sceneflow [228], classification [229], semantic segmentation [230], and 3D reconstruction [230] have all benefited from the use of synthetic data.

To the best of our knowledge, there is no method available in image captioning which use synthetic images. Existing image caption generators are only trained on labelled real images. It is important to develop caption generators for synthetic images as well. In this work, we extend the training of caption generators by using both real and synthetic images. Getting new synthetic images with appropriate

caption-labels is a challenge. To generate new synthetic but labelled images we resort to the ground truth captions available with current datasets. For example, each image in MSCOCO dataset usually has five captions. We use these captions to generate five synthetic images. We subsequently label these synthetic images with the respective captions. We use an attention-based GAN mechanism in the process to generate synthetic images. In this paper, we investigate and analyze image captioning for real images as well as machine-generated synthetic images. This paper has the following key contributions:

- We use a GAN-based text-to-image synthesis method to generate synthetic images from text.
- We use both real and synthetic images for training and testing our model.
- Finally, we demonstrate that synthetic data can significantly improve the performance of caption generators.

We organize the rest of the paper as follows: In Section 6.2, we discuss the related work. The architecture and methodology of the proposed technique are described in Section 6.3. Experimental results are discussed in Section 6.4. Section 6.5 concludes the paper.

6.2 Related Work

With the advancements in deep neural network models, automatic image captioning has become a promising research area. Hossain *et al.* [188] present a comprehensive survey of the topic. They group the methods into several categories namely, template-based image captioning, retrieval-based image captioning, and novel caption generation. Template-based methods [51] use fixed templates with a number of blank slots to generate captions. In these methods, different objects, attributes, and actions are detected first, and then the blank spaces in the templates are filled. However, templates are predefined and cannot generate variable-length captions.

Captions can also be retrieved from visual space and multi-modal space [69]. In retrieval-based methods, captions are retrieved from a set of existing captions [1]. These methods produce generalized syntactically correct captions. However, they have limitations in producing image-specific syntactically correct captions [5].

Novel captions can be generated from both visual space and multimodal space [28][29]. A typical method of this category analyzes the visual content of the image first and then generates the image captions using a language model. These methods can generate image captions that are semantically more accurate than the aforementioned approaches [5]. Most methods of this category use an encoder-decoder architecture to generate image captions [28]. In these methods, a vanilla CNN is used as the encoder to extract the image representations and an LSTM is used as a decoder to generate captions using these representations. However, these methods have problems in identifying prominent objects of the image.

Attention-based methods [29][231] can represent the prominent objects in captions because they selectively focus on the relevant objects of an image. Therefore, we use an attention-based method to generate a description of an image.

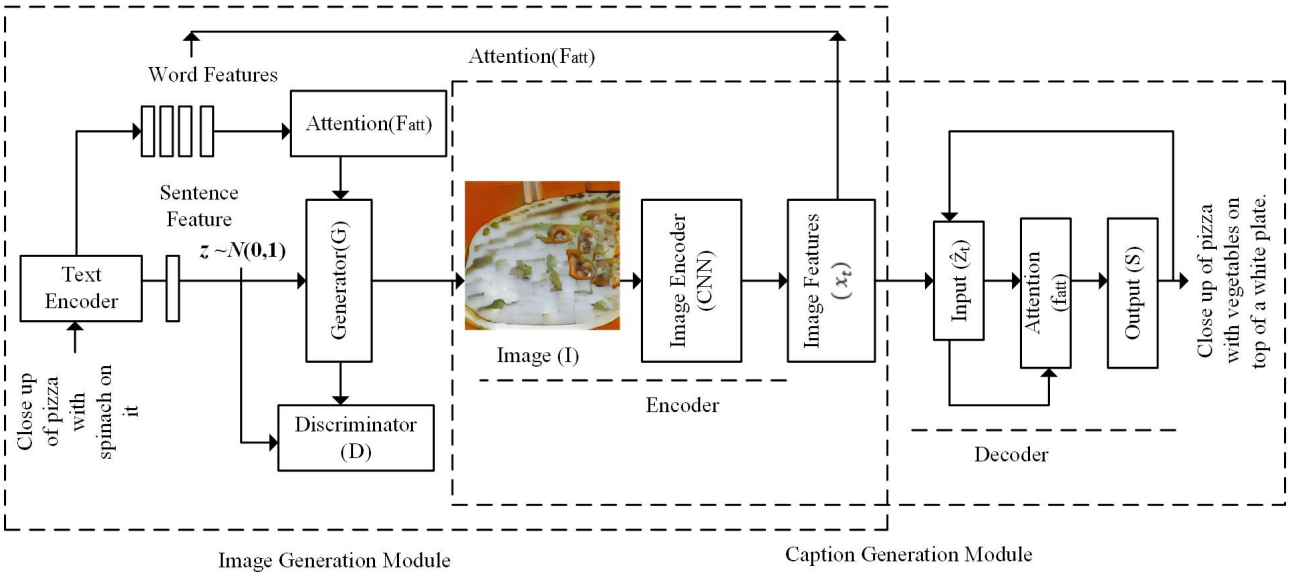


Figure 6.1: The architecture of our proposed method: a GAN-based model is used to generate synthetic images from text. The model applies attention to focus on the relevant word vectors to generate different regions of the image. Then an attention-based image captioning model is used to generate captions for that image. Image (I) can refer to any image (either real or synthetic), whichever is being used for training.

These deep learning-based image captioning methods popularly use three common publicly available datasets i.e., MSCOCO [153], Flickr30k [152], and Flickr8k [60] for training and testing the networks. These datasets were collected and annotated by humans. However, deep learning-based methods have some issues to work with these data.

- These methods require a large and diverse set of data to learn the visual representations.
- Existing models overfit the common objects that co-occur in a common context. For example, if a model is trained for a scene which contains a bed and bedroom but it is tested on unseen contexts e.g., bed and forest. The model will struggle to generalize to these scenes.
- The Manual labelling of large volume of data is expensive, biased, and time-consuming.

Synthetic data can be an attractive alternative to address these issues. To the best of our knowledge, there is no method available in image captioning which uses GAN for image synthesis. However, a number of methods [232][228] have been proposed to generate synthetic images for different computer vision tasks such as semantic segmentation, object classification, and 3D reconstruction. In recent years, GAN-based methods have shown significant advances in image synthesis. They can generate more accurate, more semantically consistent results than traditional methods. GANs can produce textured details and realistic content of an image. They are useful for many applications, such as texture synthesis, super-resolution, and image inpainting. In this paper, we use an attention-based GAN [233] for synthetic image generation.

6.3 Model Architecture

Synthetic images are used for many deep learning-based applications for training. They are used for modeling various deep learning-based methods. In this paper, we propose a pipeline whose goal is to use both real and synthetic images to train and test an image captioning method. We use an automatic system to generate synthetic images. Generative Adversarial Network (GAN) has the popularity to be used for generating realistic synthetic images. To achieve our goal, we built a pipeline composed of a GAN Module to generate synthetic images and an image captioning module to generate captions. The overall architecture of our proposed method is shown in Figure 6.1.

6.3.1 GAN Module for Synthetic Image Generation

The GAN Module learns to generate synthetic images from an input text. In this method, we use AttnGAN [233] to generate synthetic images. AttnGAN has m generators $(G_0, G_1, \ldots, G_{m-1})$. They take the hidden states $(h_0, h_1, \ldots, h_{m-1})$ as input and then generate images of different scales, from small to large $(\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{m-1})$. Therefore,

$$\begin{aligned} h_0 &= F_0(z, F^{ca}(\bar{e})), \\ h_i &= F_i(h_{i-1}, F_i^{attn}(e, h_{i-1})), \\ \hat{x}_i &= G_i(h_i) \end{aligned} \tag{6.1}$$

where, z is a latent variable which is calculated from a standard normal distribution, e is word vector matrix, $\bar{e}$ is the sentence vector. F^{ca}, F_i, F_i^{attn}, and G_i are neural networks. The attention module takes two inputs: the image features from the previous hidden state and the word features. $e \in \mathbb{R}^{D \times T}$ and $h \in \mathbb{R}^{\hat{D} \times N}$ represent the word features and the image features from the previous hidden state, respectively. First, a multi-layer perceptron is used to transfer the word features into a common semantic space. Then based on the previous hidden state features h, a word context vector is computed to generate a region of an image. The context vector can be defined as:

$$\hat{c_j} = \sum_{i=0}^{T-1} \beta_{j,i} \acute{e}_i, \text{where } \beta_{j,i} = \frac{\exp(\acute{s}_{j,i})}{\sum_{k=0}^{T-1} \exp(\acute{s}_{j,k})} \tag{6.2}$$

In the above equation, $\beta_{j,i}$ represents the weight that the model uses to attend to the i^{th} word when it generates the j^{th} region of the image. In order to generate images at the next state, the image features and the corresponding word features are combined. Both the sentence level and the word level conditions are checked to generate the final synthetic image. The module has multiple stages to generate synthetic images. Initially it generates low-resolution images. Then high-resolution images are obtained by refining the low-resolution images in multiple steps through multiple generators and discriminators. The architecture of this network is similar to a tree structure. Different branches of the tree generate images of different resolutions: at branch i, the generator G_i learns the image distribution p_{G_i} at that scale, while the discriminator D_i estimates the probability of a sample being real. The discriminator D_i takes a real image x_i or a fake sample s_i as input and is trained to classify them as

real or fake by minimizing the cross-entropy loss:

$$\mathcal{L}_{D_i} = \underbrace{-\mathbb{E}_{x_i \sim p_{data_i}}[logD_i(x_i)] - \mathbb{E}_{x_i \sim p_{G_i}}[log(1-D_i(s_i))]}_{\text{unconditional loss}} + \underbrace{-\mathbb{E}_{x_i \sim p_{data_i}}[logD_i(x_i,c)] - \mathbb{E}_{x_i \sim p_{G_i}}[log(1-D_i(s_i,c))]}_{\text{conditional loss}} \tag{6.3}$$

where x_i is an image from the true image distribution p_{data_i} at the i^{th} scale, s_i is from the model distribution p_{G_i} at the same scale. StackGAN-v2 trains a text encoder [234] following the approach of Reed et al. [235]. The encoder is used to extract visually-discriminative text embeddings of the given description. Sentences that share semantic and syntactic properties are mapped to corresponding vector representations. The multiple discriminators and generators are trained to jointly approximate multi-scale image distributions $p_{data_0}, p_{data_1}, \ldots, p_{data_{m-1}}$ by minimizing the following loss function:

$$\mathcal{L}_G = \sum_{i=1}^{m} \mathcal{L}_{G_i}, \mathcal{L}_{G_i} = \underbrace{-\mathbb{E}_{s_i \sim p_{G_i}}[logD_i(s_i)]}_{\text{unconditional loss}} + \underbrace{-\mathbb{E}_{s_i \sim p_{G_i}}[logD_i(s_i,c)]}_{\text{conditional loss}} \tag{6.4}$$

where L_{G_i} is the loss function for approximating the image distribution at the i^{th} scale. The unconditional loss is used to determine whether the image is real or fake. In contrast, the conditional loss is used to determine if the image and the condition match.

6.3.2 Image Captioning Module

The goal of the Image Captioning Module is to generate a natural language description of an image. Good performances for this task are obtained by learning a model which is able to first understand the scene described in the image, the objects taking part in it and the relationships between those objects, and then to compose a natural language sentence describing the whole picture. Given the complexity of such a task, it is still a challenging and open problem in the fields of NLP and computer vision. In our pipeline, we implement Image Captioning Module in a similar way as the one proposed in [204], meaning that we also use an attention-based captioning method based on FC models. Traditional convolutional networks with L layers have L connections. However, DenseNet has $L(L+1)/2$ direct connections. As a result, the feature-maps of all preceding layers are used as inputs to the current layer, and its own feature-maps are used as inputs into all subsequent layers. The transformation function for DenseNet is:

$$I_l = H_l([I_0, I_1, \ldots, I_{l-1}]) \tag{6.5}$$

where $[I_0, I_1, \ldots, I_{l-1}]$ refers to the concatenation of the feature-maps generated in layers $0, 1, \ldots, l-1$ and $H_l(.)$ is a composite function.

The attention-based network can recompute its attention for the relevant parts of the image according to the perceived importance from LSTM. This recomputed image feature is a dynamic representation of the relevant parts of the image and is called a context vector ($\hat{z}_t$). Such a vector is computed from the annotation vector a_i defined in equation 6.6 and the attention weight (α_{ti}). The attention weight is obtained from the alignment score (e_{ti}). The score defines how well each annotation vector matches with the previous hidden state output (h_{t-1}) of the LSTM decoder. Such an alignment

score is computed by applying an attention function (f_{att}):

$$e_{ti} = f_{\text{att}}(a_i, h_{t-1}) \tag{6.6}$$

Next, the attention weight is obtained by normalizing e_{ti} using a Softmax function:

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})} \tag{6.7}$$

Then we compute the context vector ($\hat{z}_t$) using equations 6.6 and 6.7 as follows:

$$\hat{z}_t = \phi(\{a_i\}, \{\alpha_i\}) \tag{6.8}$$

We use soft attention [29] in our experiments, where (α_i) is first computed for each image region (x_i) and then the weighted average for (x_i) is calculated to use it as an input of LSTM. Hence the context vector $\hat{z}_t$ for soft attention can be written as:

$$E_{p(x_t|a)}[\hat{z}_t] = \sum_{i=1}^{L} \alpha_i, a_i \tag{6.9}$$

Finally, the LSTM is trained to compute the output word (s_t) probability condition on the context vector ($\hat{z}_t$) and the previously generated word s_{t-1} at time t. It is defined as:

$$P(s_0, s_1, \ldots, s_m) = \prod_{i=0}^{m} P(s_i|\hat{z}, s_0, s_1 \ldots, s_m) \tag{6.10}$$

6.4 Experiments

In this section, we present the results of our experiments involving the proposed pipeline. Our pipeline has two main modules: (i) Text to Image synthesis and (ii) Image caption generation.

6.4.1 Dataset and Experimental Setup

Dataset: We use the large and popularly used MSCOCO dataset. This dataset consists of $82,783$ training and $40,504$ validation images. In our experiments we consider them as real images. In addition to these images, we also used synthetic images for our experiments. Image captioning datasets (e.g., MSCOCO that we used) have separate benchmark sets of images for training and testing. Each image has multiple ground truth captions. We used these captions to generate labelled synthetic images. We explicitly maintained the train and test split as marked in the dataset, i.e., if a synthetic image was generated from training image's ground-truth caption, that synthetic image was used in training only. On the other hand, if a synthetic image was generated from test image's ground-truth caption, that synthetic image was used in testing only.

Implementation Details: For text to image generation we follow the implementation details of AttnGAN[233]. Two neural networks: (i) text encoder and (ii) image encoder are used here. A bi-directional LSTM model [236] is used to extract the semantic vectors from the given text descriptions. Thus each word gets the context of two hidden states, one for the forward direction and one for the

backward direction. These two hidden state vectors are concatenated to compute the overall context. The feature matrix for all words are computed by $e \in \mathbb{R}^{D \times T}$, where e_i represents the feature vector of i^{th} word. D and T indicates the dimension of the vector and the total number of words, respectively. A CNN, Inception-v3 model [237] is used to extract the image feature vectors. Two types of Features namely, (i)local features of different image regions and (ii) global features of the image are extracted from the intermediate layer and the last average pooling layer, respectively. The size of the local feature matrix is $e \in \mathbb{R}^{768 \times 289}$. Here, 768 denotes the dimension of the local feature vector and 289 represents the number of sub-regions in the image. On the other hand, the size of the global feature vector is $f \in \mathbb{R}^{2048}$. Finally, a perceptron layer is used to map the image features to the semantic space of the text features.

For the image in the captioning module, we use DenseNet121 [17] with fully connected layers to extract image features. DenseNet121 is pre-trained on ImageNet dataset. We apply the fc7 feature map to compute the attention features. The dimension of our feature map is 1×1024. The size of the hidden layer in the prediction module is 1024. We apply dropout, a learning rate of 0.001 and use a linear layer to obtain a 512-dimensional word embedding. We also apply Adam optimizer with a mini-batch size 16 to train the model. Text to image generation module is implemented in Pytorch and image captioning module is implemented in Tensorflow. We used an existing PyTorch code and customise it for the image generation module, while the image captioning modules were mostly built by us on TensorFlow.

Compared Models: We demonstrate our results using qualitative analysis reported in Tables 6.1 and 6.2. In both cases, we compare the different models between them and with one baseline model. In addition, we quantitatively compare our models with other state-of-the-art image captioning models such as DeepVS [1], m-RNN [69], Google NIC [28], LRCN [166], hard-ATT [29], soft-ATT [29], and ConvCap [46] The results are shown in Table 6.3.

6.4.2 Analysis of Result

We discuss and analyze both qualitative and quantitative results of the generated captions.

Qualitative Analysis: We used both real and synthetic images for the training and testing of our different models. Next, we have generated captions for these images with these models. Then we have analyzed and compared the generated captions with a baseline method and between our different models. The generated captions in Table 6.1 are only on real images. However, the models “Train-R;Test-R" and “Train-S(1);Test-R" are trained on real images and synthetic images (the synthetic images are generated from each caption #1, for its corresponding real images), respectively. Next, these synthetic images together with the real image are used to train the model “Train-R+S(1);Test-R". Finally, all the synthetic images (the synthetic images are generated from each five captions of the corresponding real images) together with the real images are used to train the “Train-R+S(all);Test-R" model. Here, the model “Train-R;Test-R" is considered to be baseline method. It can be seen from Table 6.1 that we get longer and semantically more accurate captions when we use both real and synthetic images for training. In the first example of this table, the baseline method does not generate anything about the “jersey" of the soccer player. However, the model “Train-R+S(all);Test-R" picks this information

Input Image	Output Captions
	**Ground-Truth Captions:** Soccer player wearing red and black shirt kicking at ball. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A man playing a soccer ball on a field.* *(Train-S(1);Test-R):* *A man standing around soccer ball.* *(Train-R+S(1);Test-R):* *A man kicking a soccer ball on a soccer field.* *(Train-R+S(all);Test-R):* *A man in a soccer uniform playing soccer on a field.*
	**Ground-Truth Captions:** Woman talking on cell phone while wearing sun glasses.. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A woman holding a cell phone in her hand.* *(Train-S(1);Test-R):* *A person talking on her cell phone.* *(Train-R+S(1);Test-R):* *A woman talking on a cell phone in the sun.* *(Train-R+S(all);Test-R):* *A woman wearing sunglasses talking on a cell phone.*
	**Ground-Truth Captions:** Bowl of broccoli on cutting board. **Generated Captions:** *(Train-R;Test-R (Baseline method)):* *A bowl of food with broccoli.* *(Train-S(1);Test-R):* *A lot of broccoli sitting in bowl.* *(Train-R+S(1);Test-R):* *A plate of food with broccoli on a table.* *(Train-R+S(all);Test-R):* *A white plate of food topped with broccoli on a board.*

Table 6.1: Comparison of our different models with their generated captions on real images. The real images sample and their ground-truth captions are collected from the MS COCO dataset. 'R' means the image is from the original dataset, 'S(1)' means the synthetic images generated using the ground-truth caption 1, and 'S(all)' means the synthetic images generated from all the ground-truth captions. Images are best viewed in color.

as "uniform" successfully. Although the model "Train-R+S(1)" does not include anything about the soccer player's cloths, it picks the word "kick" which is present in the ground-truth caption. Following the example one, the "Train-R+S(all);Test-R" model successfully includes "sun glass" and "board" in the generated captions of the second and third examples, respectively. However, these words are missing in the baseline method's generated captions. Similarly, for the second and third examples, the "Train-R+S(1)" model generates captions that are closer to the ground-truth captions and these captions are semantically more accurate than the ones from the baseline method. It is also seen that the model "Train-S(1);Test-R" which is solely trained on synthetic images generates semantically weaker captions than the other models.

We illustrated the generated captions of synthetic images in Table 6.2. Since the synthetic images are very different from the real images, we do not compare the generated captions of the synthetic images with the real ones. In Table 6.2, we analyze and compare the generated captions of the synthetic images

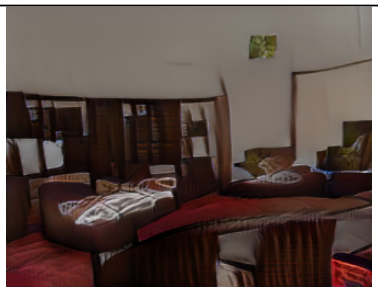
Synthetic Image	Output Captions
	**Text to Generate Image :** Pizza covered in veggies on white plate sitting on table. **Generated Captions:** *(Train-R;Test-S):* *A pizza sitting on top of a white plate.* *(Train-S(1);Test-S):* *A pizza sitting on top of a wooden table.* *(Train-R+S(1);Test-S):* *Whole pizza with slices sits on pan on the table.* *(Train-R+S(all);Test-S):* *Cheese pizza with vegetables on top of a while plate on table.*
	**Ground-Truth Captions:** Close up view of banana sitting on top of a table. **Generated Captions:** *(Train-R;Test-S):* *A bunch of bananas sitting on a table .* *(Train-S(1);Test-S):* *A bunch of banana on a table .* *(Train-R+S(1);Test-S):* *A bunch of banana that are on a table.* *(Train-R+S(all);Test-S):* *A bunch of banana sitting on a top of a wooden table.*
	**Ground-Truth Captions:** Room with bed headboard two tables and comforter. **Generated Captions:** *(Train-R;Test-S):* *A living room with a bed and a table.* *(Train-S(1);Test-S):* *A hotel room with a bed and lamp.* *(Train-R+S(1);Test-S):* *A bedroom with a blanket and pillows on it.* *(Train-R+S(all);Test-S):* *A bedroom with a white comforter with pillows and a table.*

Table 6.2: Comparison of our different models with their generated captions on synthetic images. The sample synthetic images are generated from the given text using an attention-based GAN model. 'R' means the image from the original dataset, 'S' means the synthetic images generated from the given text, 'S(1)' means the synthetic images generated using the ground-truth caption 1, and 'S(all)' means the synthetic images generated from all ground-truth captions. Images are best viewed in color.

between our different models along with the corresponding text used to generate the synthetic images. The models "Train-R+S(1)" and "Train-R+S(all)" generate reasonably better captions than other models and they are closer to the input text as well. The model "Train-R+S(all);Test-S" includes few words such as "vegetables", "top", and "white" in its generated captions of example one. It can be seen that the generated captions are longer and semantically richer than those of other models. Similarly, "top" in the second example and "white comforter", "pillows", and "table" in the third example are appropriate pick by this model. It is also seen in all three examples that the generated captions by the model "Train-R+S(1);Test-S" are semantically more accurate than those of the models "Train-R;Test-S" and "Train-S;Test-S".

Quantitative Analysis: Table 6.3 shows the results of the generated captions with our different models on BLEU-1, BLEU-2, BLEU-3, and BLEU-4 evaluation metrics. In order to demonstrate our results, we use the soft attention method proposed by Xu *et al* [29]. However, we use DenseNet instead

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGH-L	CIDEr-D
DeepVS [1]	62.5	45.0	32.1	23.0	19.5	-	66.0
m-RNN [69]	67.0	49.0	35.0	25.0	-	-	-
NIC [28]	66.6	46.1	32.9	24.6	-	-	-
g-LSTM [72]	67	49.1	35.8	26.4	23.9	-	-
Bi-LSTM-M [203]	68.7	50.9	36.4	25.8	22.9	-	73.9
LRCN [166]	69.7	51.9	38.0	27.8	22.9	50.8	83.7
Hard-ATT [29]	71.8	50.4	35.7	25.0	23.0	-	-
Soft-ATT [29]	70.7	49.2	34.4	24.3	**23.9**	-	-
ATT-FCN [41]	70.9	53.7	40.2	30.4	23.9	-	-
ConvCap [46]	69.3	51.8	37.4	26.8	23.8	51.1	85.5
COMIC [198]	70.6	53.4	39.5	29.2	23.7	**51.7**	**88.1**
Ours(Train-R;Test-R; Baseline method)	68.0	47.4	32.5	22.9	-	-	-
Ours(Train-S(1);Test-S(1))	62.7	44.4	31.1	22.0	-	-	-
Ours(Train-S(1);Test-R)	63.4	45.0	31.5	22.5	-	-	-
Ours(Train-R;Test-S(1))	66.5	46.1	34.4	23.8	-	-	-
Ours(Train-R+S(1);Test-S(1))	68.2	47.5	35.1	24.3	-	-	-
Ours(Train-R+S(1);Test-R)	71.1	53.5	40.3	30.0	-	-	-
Ours(Train-R+S(all);Test-S(all))	71.9	52.9	43.2	31.5	-	-	-
Ours(Train-R+S(all);Test-R)	**73.6**	**54.7**	**44.2**	**33.6**	-	-	-

Table 6.3: Performance of our models in comparison with other state-of-the-art techniques. Bold indicates the best results and a dash(-) indicates that results are unavailable.

of VGGNet to extract visual features from images. Xu et al. reported 70.7, 49.2, 34.4, and 24.3 scores for BLEU-1, 2,3, and 4, respectively for soft attention in their paper. However, we use the code of Yunjei available in GitHub and the scores we got are 67.7, 46.1, 32.3, and 22.4. We achieved slightly better results on DenseNet as reported in Table 6.3 and we considered it as our baseline method. In terms of BLEU scores, the models which use both real and synthetic images for training achieve superior results than other models. BLEU metrics work by counting the matching n-grams in the generated captions to the n-grams of the ground-truth captions. Therefore, It can be seen from Table 6.3 that the generated captions with some of our models can match better than the baseline method and some other state-of-art methods. For example, the model “Train-R+S(all);Test-R" achieves 73.6, 54.7, 44.2, and 33.6 BLEU scores and outperforms all the other methods. On the other hand, the models which use only synthetic images for training achieve poor results. For example, the model “Train-R;Test-S(1)" achieves 66.5, 46.1, 34.4, and 23.8 BLEU scores which are inferior to the corresponding scores of the base line method and other state-of-the-art methods.

6.5 Conclusion

We explored the use of synthetic images to generate captions from images. For this task, we built a pipeline to first generate synthetic images from text using an attention based generative adversarial network. Then we used these synthetic images together with the real images to train and test an image captioning model. We used an attention-based image captioning method to demonstrate the results. We explained the results using both qualitative and quantitative analysis. We compared the results with one baseline method and the other state-of-the-art methods. We have shown that the models, which use both real and synthetic images for training achieve superior performances compared to the baseline method and other state-of-the-art methods. In some cases, the quality of the generated synthetic images is not so good.

Chapter 7

Conclusions

This chapter presents a summary of our contributions in this thesis. We also list a number of potential directions for future work on image caption generation and related problems.

7.1 Summary

We have given an Introduction to the research problem of image captioning and a Literature Review for this thesis in **Chapters** 1 and 2, respectively. We have presented four novel methods for generating textual descriptions for images in the subsequent chapters. **Chapter** 3 has introduced an approach to generate high quality image captions that can include correct and relevant objects information. **Chapter** 4 has presented an automatic image captioning technique that include past and future contexts. **Chapter** 5 has considered a method that can generate image captions with comprehensive information namely with past, future, and local contexts. **Chapter** 6 has provided captions for synthetic images. Specifically, it has demonstrated the usefulness of synthetic images for improved captioning of real images. We have justified our methods by demonstrating empirical results using both qualitative and quantitative analysis.

In **Chapter** 1, we have presented the overview, importance, and the main research challenges of image captioning. We have also presented the aims and objectives following our contributions in this thesis.

In **Chapter** 2, we have reviewed deep learning-based image captioning methods with a taxonomy diagram. We have group them into a number of categories with their generic block diagram. We have also highlighted their pros and cons. We have presented a discussion on commonly used evaluation metrics and datasets with their strengths and weaknesses. A brief summary of experimental results is also given.

In **Chapter** 3, we have developed an image captioning framework that uses DenseNet, a type of CNN as an image encoder to extract the image features and an LSTM as a language decoder to generate captions for that image. We have also used an attention mechanism in this framework. DenseNet can extract rich image feature maps and attention mechanism can selectively focus on relevant image features. We have demonstrated its performance on the MSCOCO dataset and commonly used evaluation metrics such as BLEU, METEOR, ROUGE, and CIDEr. We have reported that our proposed method achieved better results compared to all the other methods on BLEU-2, 3, and 4 metrics and third best result on

BLEU-1. We have also compared the generated captions by our methods and VGGNet-ATT methods. We have shown that our method can generate more correct and relevant objects than VGGNet-ATT. Finally, we have illustrated the attention visualization with attention details on generated words at each time step.

In **Chapter** 4, we have developed a method that uses the past and the future context in image captioning. We have used bi-directional self attention (Bi-SAN) as language decoder to extract past and future context information. Bi-SAN applies forward and backward self-attention to obtain such contexts. We have demonstrated that the proposed Bi-SAN-based image captioning method outperforms other methods on BLEU-1, 3, 4, and ROUGH-L metrics.

In **Chapter** 5, we have extended the previous work by adding a local context together with the past and the future context for image captioning. LSTM and BLSTM have popularly been used for sequence modeling. LSTM can capture only past context. In contrast, BLSTM is capable to extract both past and future context information. However, they have limitations in capturing the local context of a sequence. CNNs can also be used as a language decoder. They perform well in preserving the local context of a sequence. Therefore, we have combined a CNN layer with a BLSTM in the language decoder in our method in **Chapter** 5. We have also used an *Attention Pooling* mechanism that can preserve significant information at the pooling stage. The combined network of a BLSTM with a CNN structure in the decoding module together with the *Attention Pooling* mechanism at the pooling stage can extract comprehensive information, namely the past, the future, and the local contexts of an image caption. We have demonstrated the empirical results using both quantitative and qualitative analysis. We have shown that our method can generate semantically richer and more meaningful captions than the LSTM, BLSTM, and CNN-based methods. We have also shown that our method can achieve superior results than most of the state-of-the-art methods. For example, our method can achieve the second best results on all the evaluation metrics except BLEU-1 over all the listed methods.

In **Chapter** 6, We have investigated the usefulness of synthetic images to generate improved image captions. We have also discussed the importance of generating captions for synthetic images. As such we have developed a pipeline that consists two modules: **(i)** text to image synthesis and **(ii)** image to text generation. We have generated synthetic images from text using an attention based generative adversarial network. These synthetic images together with the real images are used to train and test the captioning model. We have illustrated the results using both qualitative and quantitative analysis for real and synthetic images. We have also compared the results with one baseline method and the other state-of-the-art methods. We have demonstrated that the synthetic images together with the real images can generate improved image captions.

7.2 Future Work

The recent rapid progress of deep learning-based techniques opens up several avenues for further research in image captioning. We have listed here some ideas that can be investigated in the future.

In **Chapters** 3 and 4, we have explored various types of attention such as soft attention, hard attention, and self attention for image captioning. Attention mechanism has emerged as an important breakthrough for the improvement of encoder decoder-based neural networks. Investigating and

analysing other forms of attention such as spatial attention, semantic attention and combination of them can be an interesting future work.

In **Chapters** 4 and 5, we have integrated the past, the future, and the local contexts for image captioning. we have demonstrated that context is an important factor for generating semantically rich captions. There are other forms of context, for example, the context of geographic region, whether the image is being tagged or captioned, and the context of specialised knowledge. Exploring the effect of these forms of context for image captioning is an important direction for future work.

In **Chapter** 6, we have shown the effectiveness of synthetic images as data augmentation for image captioning. We have also presented the importance of generating captions for synthetic images. We used an attention-based Generative Adversarial Network for generating synthetic images. However, generating synthetic images of multiple objects is still a challenging task [233]. Therefore, further investigation needs in using other variants of GAN such as stackGAN [238], CycleGAN [239], ConditionalGAN [240] including Variational Autoencoders [241] for generating synthetic images. The scope of the **Chapter** 6 was to use synthetic images from text only. However, image synthesis from real images can be a future work. Moreover, we have covered synthetic images for improved image captioning. In future, synthetic captions for improved image captioning can be explored.

High quality captioned aligned data is not available in current datasets in vast quantities. It would be particularly helpful to exploit word substitution dictionaries from other tasks. Some authors have already attempted this [242], [243] with the machine translation. Integrating such external databases into neural network language models is still an open problem. Progress in this area has provided a direction for further exploration in image captioning.

Evaluating context in image captioning remains a challenging problem. There is no consensus on how to perform such evaluations. The traditional standard is human annotation. Even describing the target context to annotators is difficult because of the limited linguistic knowledge of many annotators on crowd-sourcing platforms. Standardising approaches to human evaluation of context-based image captions is an important direction for future work. Automatic evaluations for captions with various contexts are also necessary.

Last but not least, exploring above ideas for other forms of captioning such as visual storytelling [244], visual dialogue generation [245], and visual question answering [246] can also be an interesting avenue for further research.

Bibliography

[1] Andrej Karpathy and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3128–3137.

[2] Cheng Wang, Haojin Yang, Christian Bartz, and Christoph Meinel. "Image captioning with deep bidirectional LSTMs". In: *Proceedings of the 2016 ACM on Multimedia Conference*. ACM. 2016, pp. 988–997.

[3] Junbo Wang, Wei Wang, Liang Wang, Zhiyong Wang, David Dagan Feng, and Tieniu Tan. "Learning visual relationship and context-aware attention for image captioning". In: *Pattern Recognition* 98 (2020), p. 107075.

[4] Yue Zheng, Yali Li, and Shengjin Wang. "Intention oriented image captions with guiding objects". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 8395–8404.

[5] Raffaella Bernardi, Ruket Cakici, Desmond Elliott, Aykut Erdem, Erkut Erdem, Nazli Ikizler-Cinbis, Frank Keller, Adrian Muscat, Barbara Plank, et al. "Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures." In: vol. 55. 2016, pp. 409–442.

[6] Andrej Karpathy. "Connecting images and natural language". PhD thesis. Stanford University, 2016.

[7] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. "Gray scale and rotation invariant texture classification with local binary patterns". In: *European Conference on Computer Vision*. Springer, 2000, pp. 404–420.

[8] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection". In: *IEEE Conference on Computer Vision and Pattern Recognition* 1 (2005), pp. 886–893.

[9] David G Lowe. "Distinctive image features from scale-invariant keypoints". In: *International Journal of Computer Vision* 60.2 (2004), pp. 91–110.

[10] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[11] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. "A convolutional neural network for modelling sentences". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. 2014, pp. 655–665.

[12] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3431–3440.

[13] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *3rd International Conference on Learning Representations, ICLR*. 2015.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in Neural Information Processing Systems*. 2012, pp. 1097–1105.

[15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. “Going deeper with convolutions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*. 2015, pp. 1–9.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778.

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. “Densely connected convolutional networks”. In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2017, pp. 2261–2269.

[18] Y-Lan Boureau, Jean Ponce, and Yann LeCun. “A theoretical analysis of feature pooling in visual recognition”. In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 2010, pp. 111–118.

[19] Jürgen Schmidhuber. “Deep learning in neural networks: An overview”. In: *Neural Networks* 61 (2015), pp. 85–117.

[20] Y-Lan Boureau, Nicolas Le Roux, Francis Bach, Jean Ponce, and Yann LeCun. “Ask the locals: multi-way local pooling for image recognition”. In: *2011 International Conference on Computer Vision*. IEEE. 2011, pp. 2651–2658.

[21] Y-Lan Boureau, Francis Bach, Yann LeCun, and Jean Ponce. “Learning mid-level features for recognition”. In: *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE. 2010, pp. 2559–2566.

[22] Ehud Reiter and Robert Dale. *Building natural language generation systems*. Cambridge university press, 2000.

[23] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. “Extensions of recurrent neural network language model”. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2011, pp. 5528–5531.

[24] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780.

[25] Zhiheng Huang, Wei Xu, and Kai Yu. “Bidirectional LSTM-CRF models for sequence tagging”. In: *arXiv preprint arXiv:1508.01991* (2015).

[26] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. "Attention-based LSTM for aspect-level sentiment classification". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016, pp. 606–615.

[27] Chunting Zhou, Chonglin Sun, Zhiyuan Liu, and Francis Lau. "A C-LSTM neural network for text classification". In: *arXiv preprint arXiv:1511.08630* (2015).

[28] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. "Show and tell: A neural image caption generator". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3156–3164.

[29] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. "Show, attend and tell: Neural image caption generation with visual attention". In: *International Conference on Machine Learning*. 2015, pp. 2048–2057.

[30] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: 2014.

[31] Alex Graves and Jürgen Schmidhuber. "Framewise phoneme classification with bidirectional LSTM and other neural network architectures". In: *Neural Networks* 18.5-6 (2005), pp. 602–610.

[32] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. "Conditional image generation with pixelcnn decoders". In: *Advances in Neural Information Processing Systems*. 2016, pp. 4790–4798.

[33] Jonas Gehring, Michael Auli, David Grangier, and Yann N Dauphin. "A convolutional encoder model for neural machine translation". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 2017, pp. 123–135.

[34] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. "Convolutional sequence to sequence learning". In: *Proceedings of the 34th International Conference on Machine Learning*. 2017, pp. 1243–1252.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Advances in Neural Information Processing Systems*. 2017, pp. 5998–6008.

[36] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *Proceedings of NAACL-HLT*. 2019, 4171–4186.

[37] Maximilian Ilse, Jakub M Tomczak, and Max Welling. "Attention-based deep multiple instance learning". In: *arXiv preprint arXiv:1802.04712* (2018).

[38] Wenguan Wang and Jianbing Shen. "Deep visual attention prediction". In: *IEEE Transactions on Image Processing* 27.5 (2017), pp. 2368–2378.

[39] Pablo Barros, German I Parisi, Cornelius Weber, and Stefan Wermter. "Emotion-modulated attention improves expression recognition: A deep learning model". In: *Neurocomputing* 253 (2017), pp. 104–114.

[40] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. "Bottom-up and top-down attention for image captioning and visual question answering". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6077–6086.

[41] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. "Image captioning with semantic attention". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4651–4659.

[42] Hamed R Tavakoli and Jorma Laaksonen. "Prominent Object Detection and Recognition: A Saliency-based Pipeline". In: *CoRR* (2017).

[43] Pengfei Cao, Zhongyi Yang, Liang Sun, Yanchun Liang, Mary Qu Yang, and Renchu Guan. "Image Captioning with Bidirectional Semantic Attention-Based Guiding of Long Short-Term Memory". In: *Neural Processing Letters* 50.1 (2019), pp. 103–119.

[44] Amara Tariq and Hassan Foroosh. "Feature-independent context estimation for automatic image annotation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1958–1965.

[45] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In: *International Conference on Learning Representations*. 2015.

[46] Jyoti Aneja, Aditya Deshpande, and Alexander G Schwing. "Convolutional image captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 5561–5570.

[47] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. "Show and tell: Lessons learned from the 2015 mscoco image captioning challenge". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.4 (2017), pp. 652–663.

[48] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. "A training algorithm for optimal margin classifiers". In: *Proceedings of the fifth Annual Workshop on Computational Learning Theory* (1992), pp. 144–152.

[49] Akshi Kumar and Shivali Goel. "A survey of evolution of image captioning techniques". In: *International Journal of Hybrid Intelligent Systems* (2017), pp. 1–19.

[50] Shuang Bai and Shan An. "A Survey on Automatic Image Caption Generation". In: Elsevier, 2018.

[51] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. "Every picture tells a story: Generating sentences from images". In: *European Conference on Computer Vision*. Springer, 2010, pp. 15–29.

[52] Siming Li, Girish Kulkarni, Tamara L Berg, Alexander C Berg, and Yejin Choi. "Composing simple image descriptions using web-scale n-grams". In: *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 2011, pp. 220–228.

[53] Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. "Baby talk: Understanding and generating image descriptions". In: *Proceedings of the 24th CVPR*. Citeseer, 2011.

[54] Ahmet Aker and Robert Gaizauskas. "Generating image descriptions using dependency relational patterns". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics. 2010, pp. 1250–1258.

[55] Desmond Elliott and Frank Keller. "Image description using visual dependency representations". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 2013, pp. 1292–1302.

[56] Polina Kuznetsova, Vicente Ordonez, Alexander C Berg, Tamara L Berg, and Yejin Choi. "Collective generation of natural image descriptions". In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 2012, pp. 359–368.

[57] Polina Kuznetsova, Vicente Ordonez, Tamara L Berg, and Yejin Choi. "TREETALK: Composition and Compression of Trees for Image Descriptions." In: *TACL* 2.10 (2014), pp. 351–362.

[58] Margaret Mitchell, Xufeng Han, Jesse Dodge, Alyssa Mensch, Amit Goyal, Alex Berg, Kota Yamaguchi, Tamara Berg, Karl Stratos, and Hal Daumé III. "Midge: Generating image descriptions from computer vision detections". In: *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics. 2012, pp. 747–756.

[59] Vicente Ordonez, Girish Kulkarni, and Tamara L Berg. "Im2text: Describing images using 1 million captioned photographs". In: *Advances in Neural Information Processing Systems*. 2011, pp. 1143–1151.

[60] Micah Hodosh, Peter Young, and Julia Hockenmaier. "Framing image description as a ranking task: Data, models and evaluation metrics". In: *Journal of Artificial Intelligence Research* 47 (2013), pp. 853–899.

[61] Chen Sun, Chuang Gan, and Ram Nevatia. "Automatic concept discovery from parallel text and visual corpora". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2596–2604.

[62] Yunchao Gong, Liwei Wang, Micah Hodosh, Julia Hockenmaier, and Svetlana Lazebnik. "Improving image-sentence embeddings using large weakly annotated photo collections". In: *European Conference on Computer Vision*. Springer. 2014, pp. 529–545.

[63] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. "Unifying visual-semantic embeddings with multimodal neural language models". In: *Workshop on Neural Information Processing Systems (NIPS)*. 2014.

[64] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. "Boosting image captioning with attributes". In: *IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 4904–4912.

[65] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. "Image captioning with semantic attention". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4651–4659.

[66] Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. "Multimodal neural language models". In: *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. 2014, pp. 595–603.

[67] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, and Alan L Yuille. "Explain images with multimodal recurrent neural networks". In: *CoRR* (2014).

[68] Andrej Karpathy, Armand Joulin, and Fei Fei F Li. "Deep fragment embeddings for bidirectional image sentence mapping". In: *Advances in Neural Information Processing systems*. 2014, pp. 1889–1897.

[69] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. "Deep captioning with multimodal recurrent neural networks (m-rnn)". In: *International Conference on Learning Representations (ICLR)*. 2015.

[70] Xinlei Chen and C Lawrence Zitnick. "Mind's eye: A recurrent visual representation for image caption generation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 2422–2431.

[71] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, and John C Platt. "From captions to visual concepts and back". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1473–1482.

[72] Xu Jia, Efstratios Gavves, Basura Fernando, and Tinne Tuytelaars. "Guiding the long-short term memory model for image caption generation". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2407–2415.

[73] Junqi Jin, Kun Fu, Runpeng Cui, Fei Sha, and Changshui Zhang. "Aligning where to see and what to tell: image caption with region-based attention and scene factorization". In: 2015.

[74] ZYYYY Wu and RSWW Cohen. "Encode, Review, and Decode: Reviewer Module for Caption Generation". In: *30th Conference on Neural Image Processing System(NIPS)*. 2016.

[75] Yusuke Sugano and Andreas Bulling. "Seeing with humans: Gaze-assisted neural image captioning". In: *arXiv preprint arXiv:1608.05203* (2016).

[76] Alexander Patrick Mathews, Lexing Xie, and Xuming He. "SentiCap: Generating Image Descriptions with Sentiments." In: *AAAI*. 2016, pp. 3574–3580.

[77] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. "Densecap: Fully convolutional localization networks for dense captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4565–4574.

[78] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. "Generation and comprehension of unambiguous object descriptions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 11–20.

[79] Minsi Wang, Li Song, Xiaokang Yang, and Chuanfei Luo. "A parallel-fusion RNN-LSTM architecture for image caption generation". In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 4448–4452.

[80] Kenneth Tran, Xiaodong He, Lei Zhang, Jian Sun, Cornelia Carapcea, Chris Thrasher, Chris Buehler, and Chris Sienkiewicz. "Rich image captioning in the wild". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016, pp. 49–56.

[81] Shubo Ma and Yahong Han. "Describing images by feeding LSTM with structural words". In: *Multimedia and Expo (ICME), 2016 IEEE International Conference on*. IEEE. 2016, pp. 1–6.

[82] Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. "Dense Captioning with Joint Inference and Visual Context". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 1978–1987.

[83] Lisa Anne Hendricks, Subhashini Venugopalan, Marcus Rohrbach, Raymond Mooney, Kate Saenko, Trevor Darrell, Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, et al. "Deep compositional captioning: Describing novel object categories without paired training data". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016.

[84] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. "Knowing when to look: Adaptive attention via A visual sentinel for image captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 3242–3250.

[85] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, and Tat-Seng Chua. "SCA-CNN: Spatial and Channel-wise Attention in Convolutional Networks for Image Captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6298–6306.

[86] Zhe Gan, Chuang Gan, Xiaodong He, Yunchen Pu, Kenneth Tran, Jianfeng Gao, Lawrence Carin, and Li Deng. "Semantic compositional networks for visual captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1141–1150.

[87] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek. "Areas of Attention for Image Captioning". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 1251–1259.

[88] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li. "Deep Reinforcement Learning-based Image Captioning with Embedding Reward". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1151–1159.

[89] Cesc Chunseong Park, Byeongchang Kim, and Gunhee Kim. "Attend to You: Personalized Image Captioning with Context Sequence Memory Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6432–6440.

[90] Yufei Wang, Zhe Lin, Xiaohui Shen, Scott Cohen, and Garrison W Cottrell. "Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition". In: *Proceedings of the International Conference on Computer Vision and Pattern Recognition*. 2017.

[91] Chenxi Liu, Junhua Mao, Fei Sha, and Alan L Yuille. "Attention Correctness in Neural Image Captioning". In: *AAAI*. 2017, pp. 4176–4182.

[92] Chuang Gan, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. "Stylenet: Generating attractive visual captions with styles". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3137–3146.

[93] Bo Dai, Dahua Lin, Raquel Urtasun, and Sanja Fidler. "Towards Diverse and Natural Image Descriptions via a Conditional GAN". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2989–2998.

[94] Rakshith Shetty, Marcus Rohrbach, Lisa Anne Hendricks, Mario Fritz, and Bernt Schiele. "Speaking the Same Language: Matching Machine to Human Captions by Adversarial Training". In: *IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 4155–4164.

[95] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. "Improved image captioning via policy gradient optimization of spider". In: *Proceedings of the IEEE Internationa Conference on Computer Vision (ICCV)*. Vol. 3. 2017, pp. 873–881.

[96] Jiuxiang Gu, Gang Wang, Jianfei Cai, and Tsuhan Chen. "An empirical study of language cnn for image captioning". In: *Proceedings of the International Conference on Computer Vision (ICCV)*. 2017, pp. 1231–1240.

[97] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. "Incorporating copying mechanism in image captioning for learning novel objects". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2017, pp. 5263–5271.

[98] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. "Self-critical sequence training for image captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1179–1195.

[99] Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, Raymond Mooney, Trevor Darrell, and Kate Saenko. "Captioning images with diverse objects". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1170–1178.

[100] Li Zhang, Flood Sung, Feng Liu, Tao Xiang, Shaogang Gong, Yongxin Yang, and Timothy M Hospedales. "Actor-critic sequence training for image captioning". In: *31st Conference on Neural Information Processing Systems*. 2017.

[101] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. "Image captioning and visual question answering based on attributes and external knowledge". In: vol. 40. 6. IEEE, 2018, pp. 1367–1381.

[102] Qingzhong Wang and Antoni B Chan. "CNN+ CNN: Convolutional Decoders for Image Captioning". In: *arXiv preprint arXiv:1805.09019*. 2018.

[103] Andriy Mnih and Geoffrey Hinton. "Three new graphical models for statistical language modelling". In: *Proceedings of the 24th International Conference on Machine Learning*. ACM, 2007, pp. 641–648.

[104] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. "Exploring the limits of language modeling". In: *arXiv preprint arXiv:1602.02410* (2016).

[105] Marie-Catherine De Marneffe, Bill MacCartney, and Christopher D Manning. "Generating typed dependency parses from phrase structure parses". In: *Proceedings of LREC*. Vol. 6. Genoa Italy, 2006, pp. 449–454.

[106] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, and Tomas Mikolov. "Devise: A deep visual-semantic embedding model". In: *Advances in Neural Information Processing Systems*. 2013, pp. 2121–2129.

[107] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. "Grounded compositional semantics for finding and describing images with sentences". In: *Transactions of the Association for Computational Linguistics* 2 (2014), pp. 207–218.

[108] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets". In: *Advances in Neural Information Processing systems*. 2014, pp. 2672–2680.

[109] Ross Girshick. "Fast r-cnn". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 1440–1448.

[110] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 580–587.

[111] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards real-time object detection with region proposal networks". In: *Advances in Neural Information Processing Systems*. 2015, pp. 91–99.

[112] Chuang Gan, Tianbao Yang, and Boqing Gong. "Learning attributes equals multi-source domain generalization". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 87–97.

[113] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. "Policy gradient methods for reinforcement learning with function approximation". In: *Advances in Neural Information Processing Systems*. 2000, pp. 1057–1063.

[114] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. "Sequence level training with recurrent neural networks". In: *International Conference on learning Representations (ICLR)*. 2016.

[115] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. "Scheduled sampling for sequence prediction with recurrent neural networks". In: *Advances in Neural Information Processing Systems*. 2015, pp. 1171–1179.

[116] Vijay R Konda and John N Tsitsiklis. "Actor-critic algorithms". In: *Advances in Neural Information Processing Systems*. 2000, pp. 1008–1014.

[117] Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille. "Multi-instance visual-semantic embedding". In: *arXiv preprint arXiv:1512.06963* (2015).

[118] Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille. "Joint image-text representation by gaussian visual-semantic embedding". In: *Proceedings of the 2016 ACM on Multimedia Conference*. ACM. 2016, pp. 207–211.

[119] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*. 2017, pp. 5967–5976.

[120] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. "Generative adversarial text to image synthesis". In: *Proceedings of Machine Learning Research*. Vol. 48. 2016, pp. 1060–1069.

[121] Cristian Bodnar. "Text to Image Synthesis Using Generative Adversarial Networks". In: 2018.

[122] William Fedus, Ian Goodfellow, and Andrew M Dai. "Maskgan: Better text generation via filling in the _". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2018.

[123] Heng Wang, Zengchang Qin, and Tao Wan. "Text Generation Based on Generative Adversarial Nets with Latent Variables". In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2018, pp. 92–103.

[124] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu Seqgan. "Sequence generative adversarial nets with policy gradient." In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

[125] Eric Jang, Shixiang Gu, and Ben Poole. "Categorical reparameterization with gumbel-softmax". In: *International Conference on Learning Representations (ICLR)*. 2017.

[126] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. "The concrete distribution: A continuous relaxation of discrete random variables". In: *International Conference on Learning Representations (ICLR)*. 2017.

[127] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. "DRAW: A recurrent neural network for image generation". In: *Proceedings of Machine Learning Research*. 2015, pp. 1462–1471.

[128] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. "Spatial transformer networks". In: *Advances in Neural Information Processing Systems*. 2015, pp. 2017–2025.

[129] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. "Visual genome: Connecting language and vision using crowdsourced dense image annotations". In: *International Journal of Computer Vision* 123.1 (2017), pp. 32–73.

[130] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Advances in Neural Information Processing Systems*. 2014, pp. 3104–3112.

[131] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In: *International Conference on Learning Representations (ICLR)*. 2015.

[132] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. "On the properties of neural machine translation: Encoder-decoder approaches". In: *Association for Computational Linguistics*. 2014, pp. 103–111.

[133] Kees van Deemter, Ielka van der Sluis, and Albert Gatt. "Building a semantically transparent corpus for the generation of referring expressions". In: *Proceedings of the Fourth International Natural Language Generation Conference*. Association for Computational Linguistics. 2006, pp. 130–132.

[134] Jette Viethen and Robert Dale. "The use of spatial relations in referring expression generation". In: *Proceedings of the Fifth International Natural Language Generation Conference*. Association for Computational Linguistics. 2008, pp. 59–67.

[135] Margaret Mitchell, Kees van Deemter, and Ehud Reiter. "Natural reference to objects in a visual domain". In: *Proceedings of the 6th International Natural Language Generation Conference*. Association for Computational Linguistics. 2010, pp. 95–104.

[136] Margaret Mitchell, Kees Van Deemter, and Ehud Reiter. "Generating Expressions that Refer to Visible Objects." In: *HLT-NAACL*. 2013, pp. 1174–1184.

[137] Nicholas FitzGerald, Yoav Artzi, and Luke Zettlemoyer. "Learning distributions over logical forms for referring expression generation". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 2013, pp. 1914–1925.

[138] Dave Golland, Percy Liang, and Dan Klein. "A game-theoretic approach to generating spatial descriptions". In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. 2010, pp. 410–419.

[139] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara L Berg. "ReferItGame: Referring to Objects in Photographs of Natural Scenes." In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing(EMNLP)*. 2014, pp. 787–798.

[140] Oded Maron and Tomás Lozano-Pérez. "A framework for multiple-instance learning". In: *Advances in Neural Information Processing Systems*. 1998, pp. 570–576.

[141] Adam L Berger, Vincent J Della Pietra, and Stephen A Della Pietra. "A maximum entropy approach to natural language processing". In: *Computational Linguistics* 22.1 (1996), pp. 39–71.

[142] Franz Josef Och. "Minimum error rate training in statistical machine translation". In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2003, pp. 160–167.

[143] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "BLEU: a method for automatic evaluation of machine translation". In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 2002, pp. 311–318.

[144] Abhaya Agarwal and Alon Lavie. "Meteor, m-bleu and m-ter: Evaluation metrics for high-correlation with human rankings of machine translation output". In: *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics. 2008, pp. 115–118.

[145] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. "Cider: Consensus-based image description evaluation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 4566–4575.

[146] Jacob Devlin, Hao Cheng, Hao Fang, Saurabh Gupta, Li Deng, Xiaodong He, Geoffrey Zweig, and Margaret Mitchell. "Language models for image captioning: The quirks and what works". In: *Proceedings of the International Joint Conference on Natural Language Processing*. 2015, pp. 100–105.

[147] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. "Re-evaluation the Role of Bleu in Machine Translation Research." In: *EACL*. Vol. 6. 2006, pp. 249–256.

[148] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. "Freebase: a collaboratively created graph database for structuring human knowledge". In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. AcM. 2008, pp. 1247–1250.

[149] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. "Learning photographic global tonal adjustment with a database of input/output image pairs". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 97–104.

[150] Yahong Han and Guang Li. "Describing images with hierarchical concepts and object class localization". In: *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. ACM. 2015, pp. 251–258.

[151] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of Machine Learning Research* 3.Jan (2003), pp. 993–1022.

[152] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2641–2649.

[153] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. "Microsoft coco: Common objects in context". In: *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[154] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[155] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. "Bottom-up and top-down attention for image captioning and vqa". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6077–6086.

[156] Andreas Bulling, Jamie A Ward, Hans Gellersen, and Gerhard Troster. "Eye movement analysis for activity recognition using electrooculography". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.4 (2011), pp. 741–753.

[157] Alireza Fathi, Yin Li, and James M Rehg. "Learning to recognize daily actions using gaze". In: *European Conference on Computer Vision*. Springer. 2012, pp. 314–327.

[158] Dim P Papadopoulos, Alasdair DF Clarke, Frank Keller, and Vittorio Ferrari. "Training object class detectors from eye tracking data". In: *European Conference on Computer Vision*. Springer. 2014, pp. 361–376.

[159] Hosnieh Sattar, Sabine Muller, Mario Fritz, and Andreas Bulling. "Prediction of search targets from fixations in open-world settings". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 981–990.

[160] Karthikeyan Shanmuga Vadivel, Thuyen Ngo, Miguel Eckstein, and BS Manjunath. "Eye tracking assisted extraction of attentionally important objects from videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3241–3250.

[161] Ajay K Mishra, Yiannis Aloimonos, Loong Fah Cheong, and Ashraf Kassim. "Active visual segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.4 (2012), pp. 639–653.

[162] S Karthikeyan, Vignesh Jagadeesh, Renuka Shenoy, Miguel Ecksteinz, and BS Manjunath. "From where and how to what we see". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013, pp. 625–632.

[163] Kiwon Yun, Yifan Peng, Dimitris Samaras, Gregory J Zelinsky, and Tamara L Berg. "Exploring the role of gaze behavior and object detection in scene understanding". In: *Frontiers in Psychology* 4 (2013).

[164] Gregory J Zelinsky. "Understanding scene understanding". In: *Frontiers in Psychology* 4 (2013).

[165] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. "Salicon: Saliency in context". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1072–1080.

[166] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 2625–2634.

[167] Junhua Mao, Xu Wei, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan L Yuille. "Learning like a child: Fast novel visual concept learning from sentence descriptions of images". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2533–2541.

[168] Rémi Lebret, Pedro O Pinheiro, and Ronan Collobert. "Simple image description generator via a linear phrase-based approach". In: *Workshop on International Conference on Learning Representations (ICLR)*. 2015.

[169] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3431–3440.

[170] Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. "Multi-task sequence to sequence learning". In: *International Conference on Learning Representations (ICLR)*. 2016.

[171] Yoshua Bengio, Rjean Ducharme, Pascal Vincent, and Christian Jauvin. "A neural probabilistic language model". In: *Journal of Machine Learning Research* 3.Feb (2003), pp. 1137–1155.

[172] Andriy Mnih and Geoffrey Hinton. “Three new graphical models for statistical language modelling”. In: *Proceedings of the 24th International Conference on Machine Learning*. ACM. 2007, pp. 641–648.

[173] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient estimation of word representations in vector space”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2013.

[174] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. “Recurrent neural network based language model”. In: *Eleventh Annual Conference of the International Speech Communication Association*. 2010.

[175] Qi Wu, Chunhua Shen, Anton van den Hengel, Lingqiao Liu, and Anthony Dick. “Image captioning with an intermediate attributes layer”. In: *arXiv preprint arXiv:1506.01144*. 2015.

[176] Tseng-Hung Chen, Yuan-Hong Liao, Ching-Yao Chuang, Wan-Ting Hsu, Jianlong Fu, and Min Sun. “Show, Adapt and Tell: Adversarial Training of Cross-domain Image Captioner”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Vol. 2. 2017.

[177] Michael Grubinger, Paul Clough, Henning Müller, and Thomas Deselaers. “The iapr tc-12 benchmark: A new evaluation resource for visual information systems”. In: *International Workshop OntoImage*. Vol. 5. 2006, p. 10.

[178] Etienne Denoual and Yves Lepage. “BLEU in characters: towards automatic MT evaluation in languages without word delimiters”. In: *Companion Volume to the Proceedings of the Second International Joint Conference on Natural Language Processing*. 2005, pp. 81–86.

[179] Chin-Yew Lin and Franz Josef Och. “Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics”. In: *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 2004, p. 605.

[180] Chin-Yew Lin. “Rouge: A package for automatic evaluation of summaries”. In: *Text summarization branches out: Proceedings of the ACL-04 workshop*. Vol. 8. Barcelona, Spain. 2004.

[181] Satanjeev Banerjee and Alon Lavie. “METEOR: An automatic metric for MT evaluation with improved correlation with human judgments”. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Vol. 29. 2005, pp. 65–72.

[182] Stephen Robertson. “Understanding inverse document frequency: on theoretical arguments for IDF”. In: *Journal of Documentation* 60.5 (2004), pp. 503–520.

[183] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. “Spice: Semantic propositional image caption evaluation”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 382–398.

[184] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. “Image retrieval using scene graphs”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3668–3678.

[185] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D Manning. "Generating semantically precise scene graphs from textual descriptions for improved image retrieval". In: *Proceedings of the fourth Workshop on Vision and Language*. Vol. 2. 2015.

[186] Naeha Sharif, Lyndon White, Mohammed Bennamoun, and Syed Afaq Ali Shah. "Learning-based Composite Metrics for Improved Caption Evaluation". In: *Proceedings of ACL 2018, Student Research Workshop*. 2018, pp. 14–20.

[187] Jia-Yu Pan, Hyung-Jeong Yang, Christos Faloutsos, and Pinar Duygulu. "Gcap: Graph-based automatic image captioning". In: *2004 Conference on Computer Vision and Pattern Recognition Workshop*. IEEE. 2004, pp. 146–146.

[188] MD Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. "A Comprehensive Survey of Deep Learning for Image Captioning". In: *ACM Computing Surveys (CSUR)* 51.6 (2019), p. 118.

[189] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. "Imagenet large scale visual recognition challenge". In: *International journal of computer vision*. Vol. 115. 3. Springer, 2015, pp. 211–252.

[190] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *International Conference on Learning Representations*. 2014.

[191] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. "Knowing when to look: Adaptive attention via a visual sentinel for image captioning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 375–383.

[192] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. "Neural machine translation in linear time". In: *arXiv preprint arXiv:1610.10099* (2017).

[193] Xinxin Zhu, Lixiang Li, Jing Liu, Haipeng Peng, and Xinxin Niu. "Captioning transformer with stacked attention modules". In: *Applied Sciences* 8.5 (2018), p. 739.

[194] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. "Bi-directional block self-attention for fast and memory-efficient sequence modeling". In: *Proceedings of the International Conference on Language Representations (ICLR)*. 2018.

[195] Wenhao Jiang, Lin Ma, Yu-Gang Jiang, Wei Liu, and Tong Zhang. "Recurrent fusion network for image captioning". In: *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 499–515.

[196] Linghui Li, Sheng Tang, Lixi Deng, Yongdong Zhang, and Qi Tian. "Image caption with global-local attention". In: *Thirty-First AAAI Conference on Artificial Intelligence* (2017).

[197] Senmao Ye, Junwei Han, and Nian Liu. "Attentive Linear Transformation for Image Captioning". In: *IEEE Transactions on Image Processing* 27.11 (2018), pp. 5514–5524.

[198] Jia Huei Tan, Chee Seng Chan, and Joon Huang Chuah. "COMIC: Towards a compact image captioning model with attention". In: *IEEE Transactions on Multimedia* (2019).

[199] Yi Bin, Yang Yang, Fumin Shen, Ning Xie, Heng Tao Shen, and Xuelong Li. "Describing video with attention-based bidirectional LSTM". In: *IEEE Transactions on Cybernetics* 49.7 (2018), pp. 2631–2641.

[200] Yu Liu, Jianlong Fu, Tao Mei, and Chang Wen Chen. "Let your photos talk: Generating narrative paragraph for photo stream via bidirectional attention recurrent neural networks". In: *Thirty-First AAAI Conference on Artificial Intelligence* (2017).

[201] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. "Disan: Directional self-attention network for rnn/cnn-free language understanding". In: *Thirty-Second AAAI Conference on Artificial Intelligence* (2018).

[202] Sebastian Springenberg, Egor Lakomkin, Cornelius Weber, and Stefan Wermter. "Image-to-Text Transduction with Spatial Self-Attention." In: *Proceedings of the the European Symposium on Artificial Neural Networks (ESANN)*. 2018.

[203] Cheng Wang, Haojin Yang, and Christoph Meinel. "Image captioning with deep bidirectional lstms and multi-task learning". In: *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 14.2s (2018), p. 40.

[204] Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammed Bennamoun. "Attention-Based image captioning using DenseNet features". In: *Proceedings of the International Conference on Neural Information Processing*. Springer. 2019, pp. 109–117.

[205] Md Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, and Mohammed Bennamoun. "Bi-san-cap: bi-directional self-attention for image captioning". In: *Proceedings of the International Conference of Digital Image Computing: Techniques and Applications (DICTA)*. IEEE. 2019, pp. 1–7.

[206] Yiqing Huang, Jiansheng Chen, Wanli Ouyang, Weitao Wan, and Youze Xue. "Image Captioning With End-to-End Attribute Detection and Subsequent Attributes Prediction". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 4013–4026.

[207] Meng Joo Er, Yong Zhang, Ning Wang, and Mahardhika Pratama. "Attention pooling-based convolutional neural network for sentence modelling". In: *Information Sciences* 373 (2016), pp. 388–403.

[208] Rie Johnson and Tong Zhang. "Effective use of word order for text categorization with convolutional neural networks". In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2015), pp. 103–112.

[209] Omer Levy, Yoav Goldberg, and Ido Dagan. "Improving distributional similarity with lessons learned from word embeddings". In: *Transactions of the Association for Computational Linguistics* 3 (2015), pp. 211–225.

[210] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. "Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2014, pp. 238–247.

[211] Omer Levy and Yoav Goldberg. "Neural word embedding as implicit matrix factorization". In: *Advances in neural information processing systems*. 2014, pp. 2177–2185.

[212] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[213] Yoon Kim. "Convolutional neural networks for sentence classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1746–1751.

[214] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. "Rectifier nonlinearities improve neural network acoustic models". In: *Proceedings of the 30th International Conference on Machine Learning*. Vol. 30. 1. 2013, p. 3.

[215] Xiaolei Niu and Yuexian Hou. "Hierarchical Attention BLSTM for Modeling Sentences and Documents". In: *International Conference on Neural Information Processing*. Springer. 2017, pp. 167–177.

[216] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5659–5667.

[217] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. "Paying more attention to saliency: Image captioning with saliency and context attention". In: *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 14.2 (2018), pp. 1–21.

[218] Ning Xu, An-An Liu, Jing Liu, Weizhi Nie, and Yuting Su. "Scene graph captioner: Image captioning based on structural visual representation". In: *Journal of Visual Communication and Image Representation* 58 (2019), pp. 477–485.

[219] Chen He and Haifeng Hu. "Image captioning with text-based visual attention". In: *Neural Processing Letters* 49.1 (2019), pp. 177–185.

[220] Xinghan Chen, Mingxing Zhang, Zheng Wang, Lin Zuo, Bo Li, and Yang Yang. "Leveraging unpaired out-of-domain data for image captioning". In: *Pattern Recognition Letters* 132 (2020), pp. 132–140.

[221] Songtao Ding, Shiru Qu, Yuling Xi, and Shaohua Wan. "Stimulus-driven and concept-driven analysis for image caption generation". In: *Neurocomputing* 398 (2020), pp. 520–530.

[222] Ye Zhang and Byron Wallace. "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification". In: *Proceedings of the International Joint Conference on Natural Language Processing*. 2017.

[223] Naila Murray and Florent Perronnin. "Generalized max pooling". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2473–2480.

[224] Lyndon White, Roberto Togneri, Wei Liu, and Mohammed Bennamoun. *Neural Representations of Natural Language*. Vol. 783. Springer, 2018.

[225] Salman Khan, Hossein Rahmani, Syed Afaq Ali Shah, and Mohammed Bennamoun. "A guide to convolutional neural networks for computer vision". In: *Synthesis Lectures on Computer Vision* 8.1 (2018), pp. 1–207.

[226] Haiyang Wei, Zhixin Li, Canlong Zhang, Tao Zhou, and Yu Quan. "Image Captioning Based On Sentence-Level And Word-Level Attention". In: *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2019, pp. 1–8.

[227] Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige. "On pre-trained image features and synthetic images for deep learning". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 682–697.

[228] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4040–4048.

[229] Joao Borrego, Atabak Dehban, Rui Figueiredo, Plinio Moreno, and Bernardino. "Applying domain randomization to synthetic data for object category detection". In: *arXiv preprint arXiv:1807.09834* (2018).

[230] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 3234–3243.

[231] Sen He, Hamed R Tavakoli, Ali Borji, and Nicolas Pugeault. "Human Attention in Image Captioning: Dataset and Analysis". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 8529–8538.

[232] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, and Golkov. "Flownet: Learning optical flow with convolutional networks". In: *Proceedings of the IEEE international conference on computer vision*. IEEE, 2015, pp. 2758–2766.

[233] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. "Attngan: Fine-grained text to image generation with attentional generative adversarial networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1316–1324.

[234] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. "Stackgan++: Realistic image synthesis with stacked generative adversarial networks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.8 (2018), pp. 1947–1962.

[235] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. "Learning deep representations of fine-grained visual descriptions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 49–58.

[236] Mike Schuster and Kuldip K Paliwal. "Bidirectional recurrent neural networks". In: *IEEE Transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.

[237] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2818–2826.

[238] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 5907–5915.

[239] Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. "Augmented cyclegan: Learning many-to-many mappings from unpaired data". In: *arXiv preprint arXiv:1802.10151* (2018).

[240] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE Conference on Computer Vsion and Pattern Recognition*. 2017, pp. 1125–1134.

[241] Carl Doersch. "Tutorial on variational autoencoders". In: *arXiv preprint arXiv:1606.05908* (2016).

[242] Courtney Napoles, Chris Callison-Burch, and Matt Post. "Sentential paraphrasing as black-box machine translation". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. 2016, pp. 62–66.

[243] Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. "Optimizing statistical machine translation for text simplification". In: *Transactions of the Association for Computational Linguistics* 4 (2016), pp. 401–415.

[244] Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. "Visual storytelling". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2016, pp. 1233–1239.

[245] Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton Van Den Hengel. "Are you talking to me? reasoned visual dialog generation through adversarial learning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6106–6115.

[246] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. "Vqa: Visual question answering". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2425–2433.